Final Report
Office of Naval Research
Grant #N00014-03-1-0094
Systematic Procedural Error

Principal Investigator: Michael D. Byrne
Department of Psychology
Rice University
6100 Main St., MS-25
Houston, TX 77005
+1 713-348-3770 voice
+1 713-348-5221 fax
byrne@rice.edu

1. Introduction

Even in the execution of known routine procedures, people make nonrandom errors. Everyone has had this experience, whether it is leaving one's bank card in an ATM or failing to attach a promised file to an email message. While many such errors have little or no real cost, many such errors have dire consequences, including loss of human life (Casey, 1998). In many situations faced by Navy personnel, the potential consequences tend toward the severe end of this range (e.g., the *Vincennes* incident in the 1991 Gulf War). An understanding of the mechanisms underlying such errors, and therefore ultimately knowledge about how to potentially defeat them, would clearly be valuable to the Navy, other Department of Defense organizations, and private industry. Clearly, an understanding of the cognitive and perceptual mechanisms underlying such error, and therefore knowledge about how to potentially defeat them, would be valuable. This is particularly

However, this problem has spawned surprisingly little research. Senders and Moray (1991, p. 2) identify probably the major explanation: "one reason for this is that error is frequently considered only as *result* or *measure* of some other variable, and not a phenomenon in its own right." Empirical work on systematic errors in the execution of routine procedures is dominated by anecdotal accounts (e.g., Casey, 1998) but controlled experiments on this subject are quite rare. The dominant theoretical paradigm in this area is certainly the one proposed by Reason (1990), which is more a taxonomy than a theory. Reason classifies errors into two types: "mistakes," which are the result of forming an incorrect intention to act, and "slips," which are failures to correctly execute an intention. These are tied to Rasmussen's (1987) skill-rule-knowledge (SRK) hierarchy of skill acquisition. Mistakes are usually errors at the knowledge level; that is, the person making the error has incorrect knowledge about how to perform the task.

While this is certainly the explanation for many errors, it does not appear to apply to many interesting forms of systematic procedural error in which the person does know the correct set of steps. Reason generally attributes errors at the skill or rule levels of performance, to "lapses of attention." From the perspective of trying to understand the causes of such errors, this is not helpful. First, it is at best postdictive, not predictive. Second, it simply shifts the locus of the problem to another area of psychology which is, at best, ill-defined.

This research represents an effort to improve this situation within a particular domain, that of errors in the execution of routine procedures. Routine procedures are those which fall under the heading of routine cognitive skill as defined by Card, Moran, and Newell (1983) and John and Kieras (1996). Such a skill is one where the person executing the skill has the correct knowledge of how to perform the task and simply needs to execute that knowledge. Roughly speaking, that can be thought of as the point where people are no longer problem solving, but rather applying proceduralized knowledge to a relatively familiar task.

This level of skill has been the focus of attention for an entire family (the GOMS family; John & Kieras, 1996) of techniques for analysis and execution time prediction. This is largely due to the fact that such a wide array of situations fall under this classification, from occasional but not infrequent programming of VCRs to situations involving highly-motivated people in safety-critical situations, such as commercial pilots and medical professionals. As noted, GOMS, which stands for goals, operators, methods, and selection rules, is one of the primary techniques for predicting human performance under these conditions, and the empirical success of GOMS is well-documented (again, see John & Kieras, 1996). A typical GOMS analysis is based on a hierarchical goal decomposition and then a listing of the primitive operators needed to carry out the lowest-level goals. Thus, GOMS analyses are highly sensitive to the goal-based task structure and the number of primitive operations required.

However, GOMS-class analyses do not take into account visual factors of the interface such as the layout of the controls used when executing a procedure. Furthermore, the model of cognitive control underlying the GOMS approach, goal stacks, does not appear to be adequate to explain well-known error types, particularly postcompletion errors. (Postcompletion errors are errors in which the operator omits a step or subgoal of the procedure where that step or subgoal occurs after the main goal of the task is completed. Examples include leaving a bankcard in an automated teller machine or driving off without the gas cap after filling the tank.) While GOMS analyses can identify where postcompletion errors might occur, they do not explain why some cue-based mitigation strategies are effective and why others are not.

Issues of cognitive control, and in particular, cognitive control of vision, are likely to become increasingly important as more and more interfaces become visual and visual interfaces are deployed more widely (such as cell phones, PDAs, and in-car navigation systems). There are few researchers working at the boundary between vision and cognition and even fewer who are also concerned with how such performance impacts how people make errors.

Since GOMS is the dominant tool for understanding human performance in such tasks, the fact that it cannot accommodate these results is significant for anyone who wants to predict or explain how people execute routine procedures. Because errors (and sometimes even simple slowdowns) in the execution of routine procedures can have such a high cost, it is important to have not only a thorough understanding of such effects, but ultimately to have a model which predicts how people will perform when executing routine procedures.

2. Common Experimental Methods

Multiple experiments were completed in this funding period. All of these experiments have as their basis a common set of experimental tasks and methods, so these common methods will be described in some detail so the reader may become familiar with them. These methods are derived from the methods used in Byrne and Bovair (1997).

There were two primary tasks involved, set in a fictional "Star Trek" setting to engage experimental participants. (Rice University, where the experiments were conducted, has a strong science and engineering presence and thus this was indeed an engaging cover story.) Both tasks were, as described, routine procedures which the participants simply had to memorize. Each procedure was broken into subgoals on which participants were explicitly instructed. Recall that GOMS analysis represents the state-of-the-art in terms of task analysis for routine procedural tasks. What such an analysis predicts is that two tasks with the same goal/method/operator structure should produce identical performance. These two tasks had the same basic goal/method/operator structure and are thus termed "GOMS-isomorphic."

The subgoals and steps in each task are listed in Table 1 and the displays for each task are presented in Figure 1. Participants were trained to a performance criterion (four error-free trials) on each task in the first experimental session and then returned approximately one week later for a second session. During the second session, the experiment program emitted warning beeps on error commission. A concurrent working memory letter task was also introduced on the day of testing. As in the study by Byrne and Bovair (1997), its function was to increase working memory load during task performance. Participants were presented with auditory stimuli in the form of randomly ordered letters spoken through the headphones at a rate of one letter every three seconds. A tone was presented randomly at intervals ranging from nine to forty-five seconds upon which the participants were directed to recall the last three letters in order and type them into the text box that appeared on the screen.

Participants were encouraged to work both accurately and quickly by means of a scoring system, an onscreen timer, and prizes. The scoring system incremented points for each correctly executed step and decremented points for each incorrect. Bonus points were awarded for task completion within a set time. The exact scoring scheme used varied slightly from experiment to experiment. The number of trials of each task completed in the second session also varied slightly from experiment to experiment; most were in the range of 12–14 times per task.

Table 1. Subgoals and steps for the Phaser and Transporter tasks used in the experiments.

Step #	**Phaser**	**Transporter**
First subgoal		
1	Power Connected	Scanner On
2	Charge	Active Scan
3	Stop Charging	Lock Signal
4	Power Connected	Scanner Off
Second subgoal		
5	Settings	Enter Frequency
6	<slider>	<type>
7	Focus Set	Accept Frequency
Third subgoal		
8	Firing	Transporter Power
9	Tracking	Synchronous Mode
10	<track-and-space>	<track-and-click>
11	Tracking	Synchronous Mode
12	*Main Control*	*Main Control*

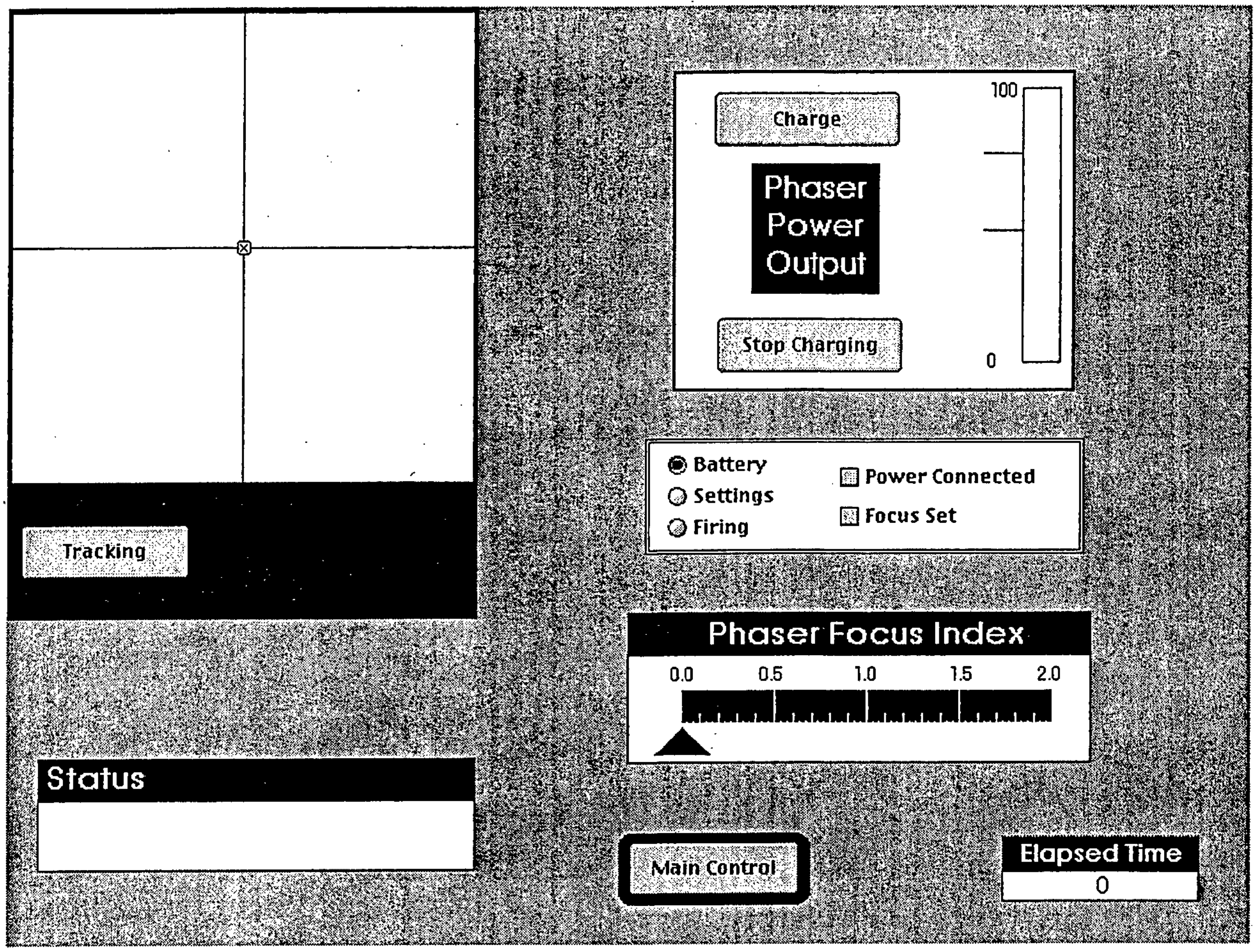


Figure 1a. Task display for the "phaser" task.

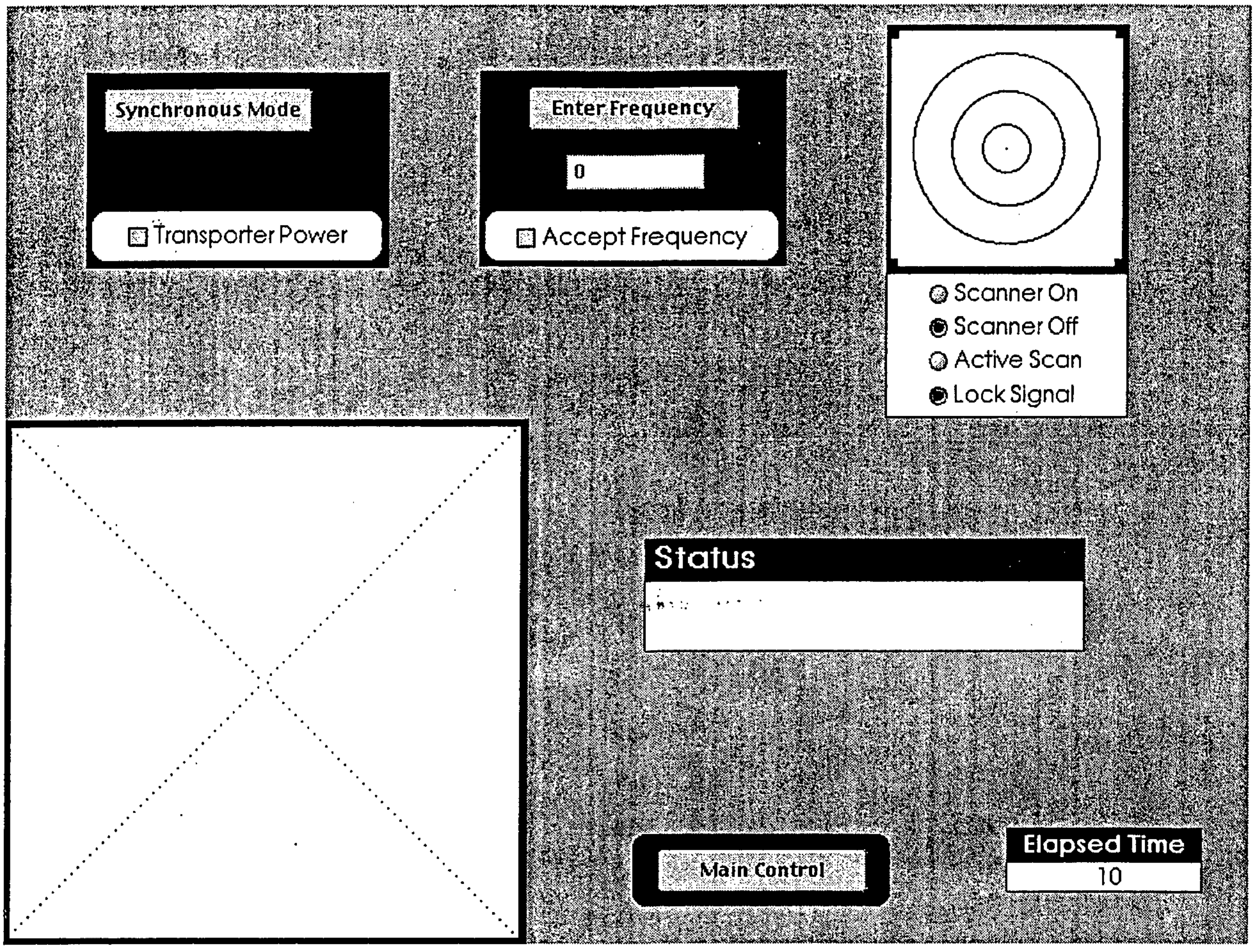


Figure 1b. Task display for the "transporter" task.

There were two primary measures of performance, error frequency and step completion time. it was possible to have several opportunities within a single trial to commit an error at each action step. Error frequency is defined as the number of errors at step X_i divided by the number of opportunities for error at step X_i. Each step can be considered a sequential choice (Ohlsson, 1996), so the definition was based on the step, not the action. That is, each step was counted as either containing an error or not containing an error, regardless of the number of incorrect actions taken. For example, if a participant was at step 3 of the procedure and clicked one incorrect button, that means an error was made at at step 3. Further incorrect clicks made there do not advance the state of the procedure, so they were not counted as additional errors. This focuses the analysis on where in the procedure the errors occurred. Step completion time was measured at the time elapsed (in milliseconds) between the successful execution of the previous step and the successful execution of the current step, that is, an inter-click latency. Steps containing errors were omitted from this analysis.

3. Major Findings

3.1 Empirical Results: Layout Error

The first important question was whether or not two tasks which are GOMS-isomorphic could generate markedly different performance in terms of both execution time and error rates. Figure 2, which is based on the aggregate from 3 different experiments, shows how two GOMS-isomorphic tasks can differ in error frequency, and Figure 3 shows the differences in step completion time. (Note that not all steps are included in the step completion time analysis because some of the times are not simple inter-click latencies but include other activities such as waiting for the interface or performing a tracking task.)

These results are quite conclusive as to the fact that the tasks did indeed have different performance profiles, though they did not allow a clear assessment as to why the two tasks were different. Further research, however, was able to provide insights into this difference.

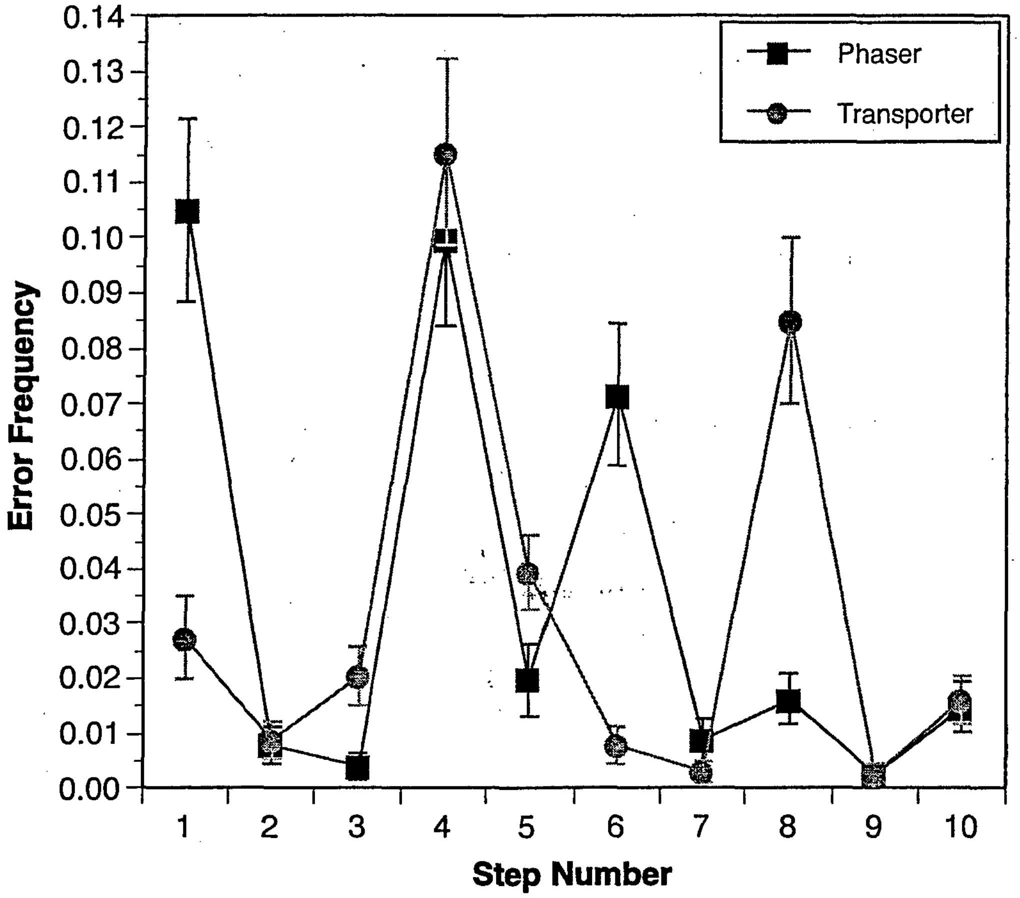


Figure 2. Error frequency as a function of step number in two GOMS-isomorphic tasks (Phaser and Transporter). Error bars depict 95% confidence intervals based on 164 subjects.

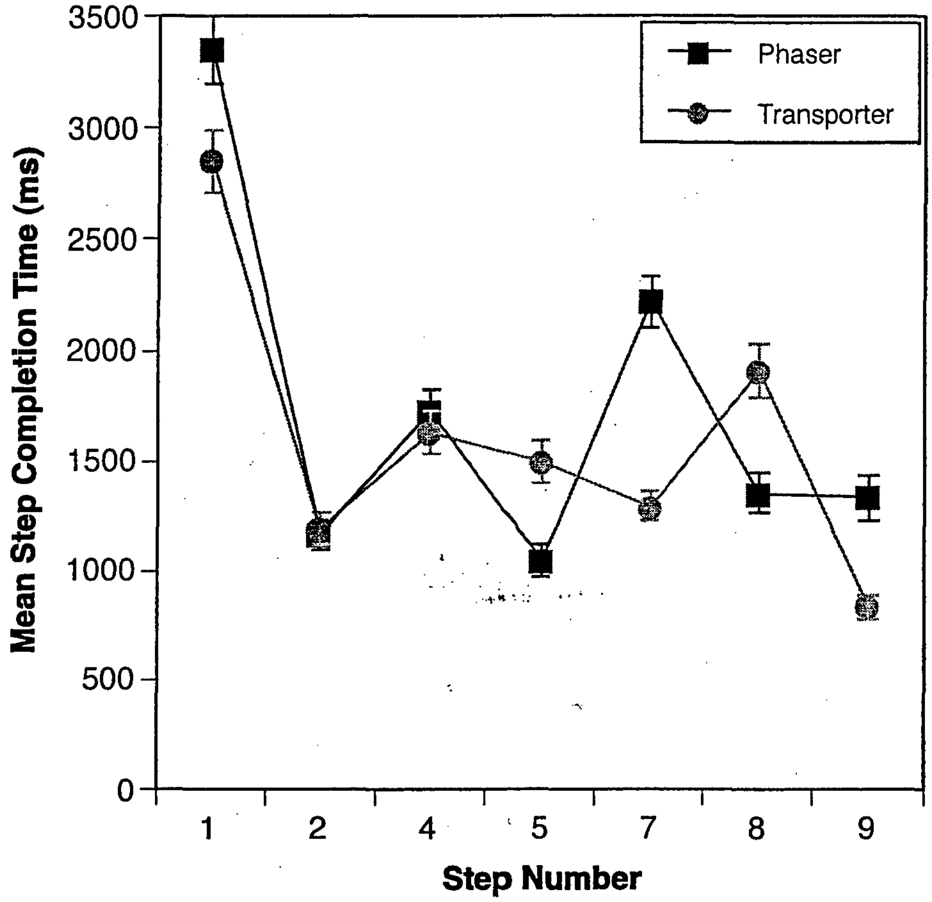


Figure 3. Step completion time as a function of step number in two GOMS-isomorphic tasks (Phaser and Transporter). Error bars depict 95% confidence intervals based on 164 subjects.

Human performance on these tasks is highly sensitive to the layout of the controls (e.g., buttons, sliders) on the display. The initial results were based on two tasks isomorphic in GOMS structure but differing in control layout (and cover story). Subsequent experiments explicitly manipulated control layouts and yielded the following discoveries:

First, visual grouping makes a substantial difference in performance. Grouping controls according to the organization of subtasks clearly yields superior performance to grouping based on control type (i.e., all radio buttons in one group, all pushbuttons in another, etc.). See results presented in Figures 4 and 5, which clearly show that error rate and task execution rate are affected by how the controls on the display are visually grouped. Note that in a replication study of the "Task" grouping, the error rate for step 6 was substantially reduced, providing further evidence that visual grouping interacts with task structure.

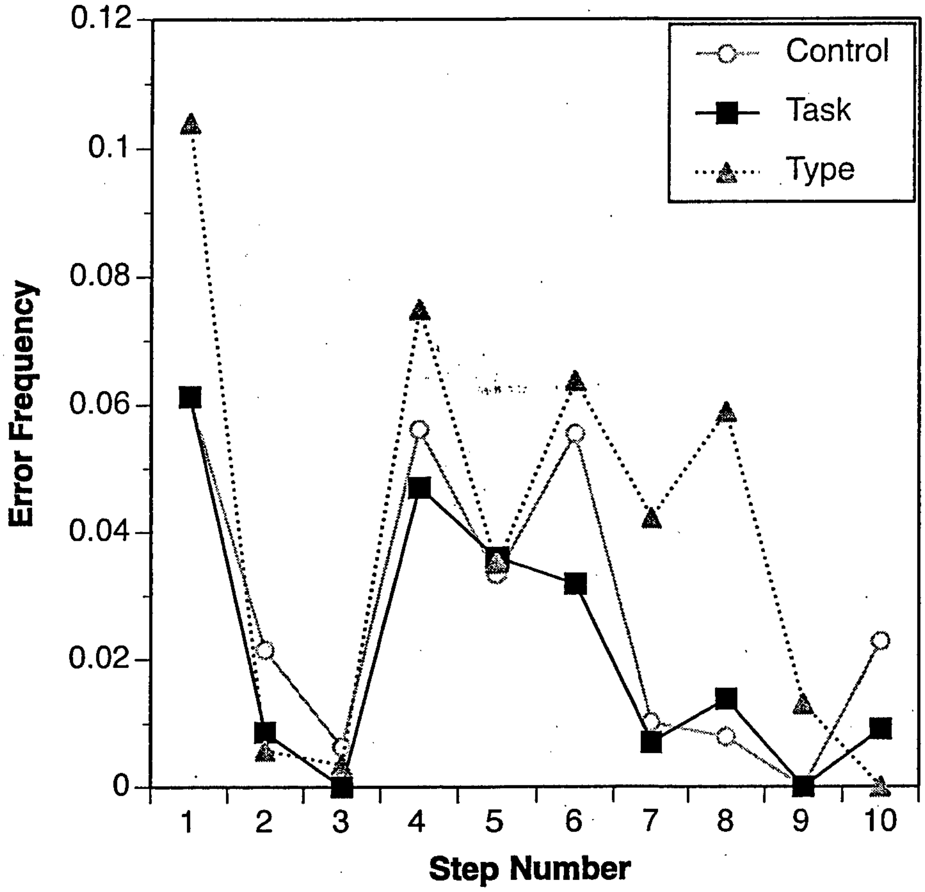


Figure 4. Error frequency for different control layouts. Control = original layout; Task = controls goruped by subtask; Type = controls grouped by control type.

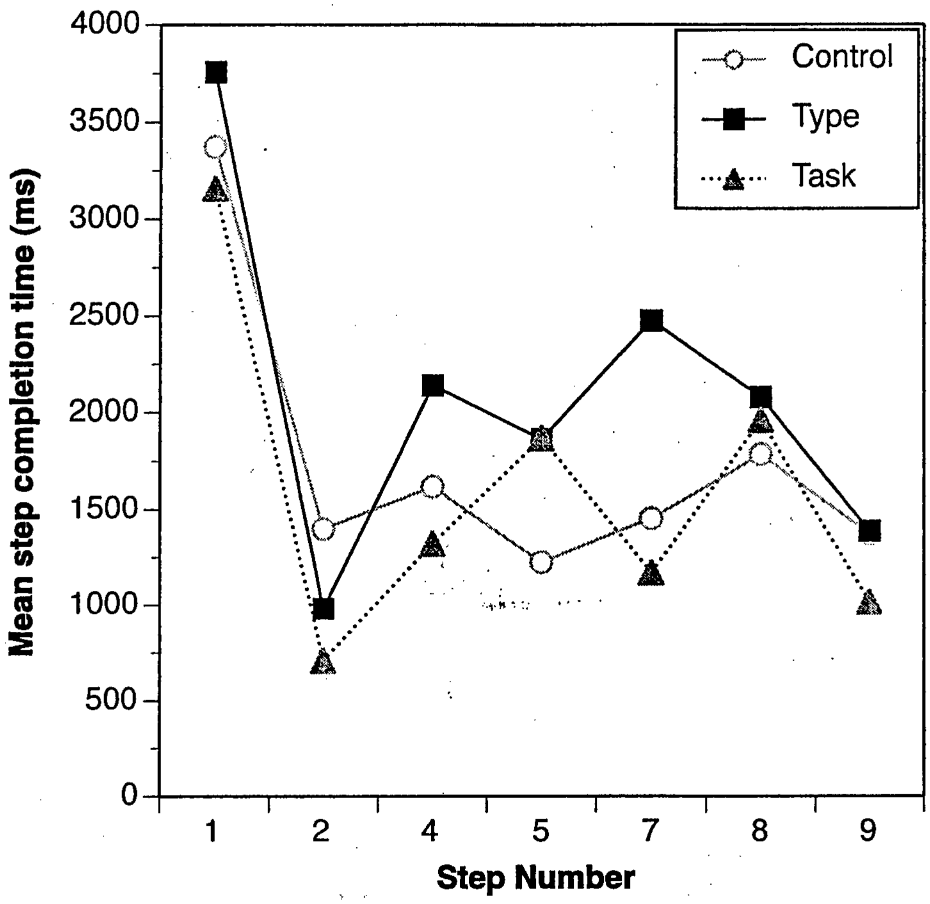


Figure 5. Step completion time for different control layouts. Control = original layout; Task = controls goruped by subtask; Type = controls grouped by control type.

Second, users are sensitive to both global and local constraints on where controls "should" be. That is, they expect consistency with global considerations like reading order. However, they are also sensitive to local considerations like how controls have been organized in other parts of the display. Violating either constraint can lead users to err. See results presented in Figure 6 which depicts error rate at a particular step in one of the procedures (step 8 in the Transporter; see Figure 2 to see how this compares to other steps). In this experiment, two of the conditions were inconsistent with expectation, violating either local or global consistency. In the two other conditions, this step was consistent with global expectations and the local ordering of the prior subgoal ("subgoal consistent" condition) or with all prior subgoals ("full consistent" condition). Clearly, both constraints must be honored in order to reduce error rates to subsystematic frequency (i.e., less than 5%).

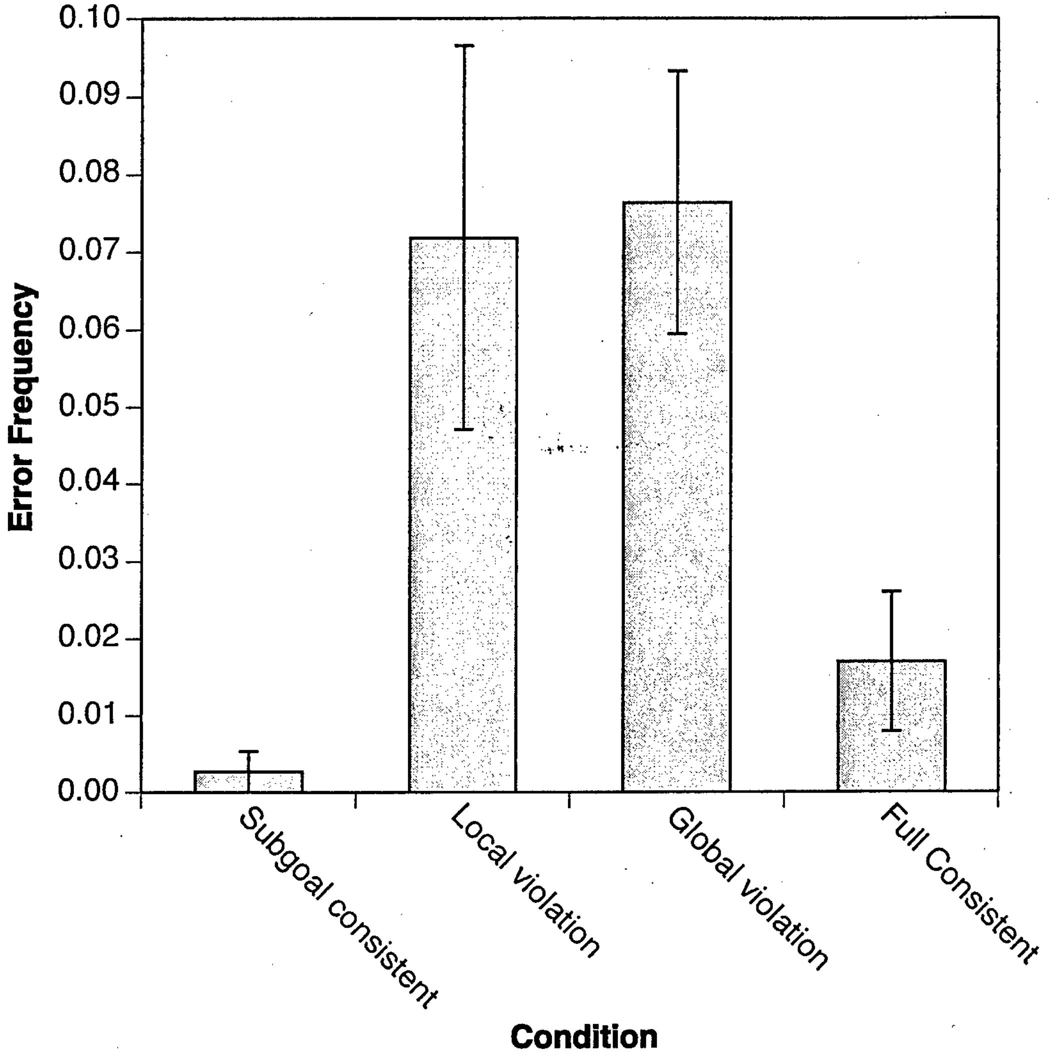


Figure 6. Error rate at step 8 in the Transporter task by condition. See text for further explanation. Error bars represent one standard error of the mean.

Performance is surprisingly insensitive to the surface features of the controls themselves or the number of extraneous controls. Users apparently do not use the local state of the controls (e.g., checked state of checkboxes) to track task progress in such routine tasks. Similarly, adding many extraneous controls has little impact on performance. See results presented in Figure 7, which shows step completion time for the control version of the Phaser task along with two variants, the "extra buttons" variant and the "push buttons" variant. In the "extra buttons" condition, numerous extraneous buttons were added to the display. In the "push buttons" condition, all the buttons were converted into push buttons rather than button types which display state information (e.g., radio buttons or checkboxes). Similar results were obtained for the Transporter task. It should be noted that GOMS models have nothing to say about any effects of additional buttons or a lack of state information on the interface.

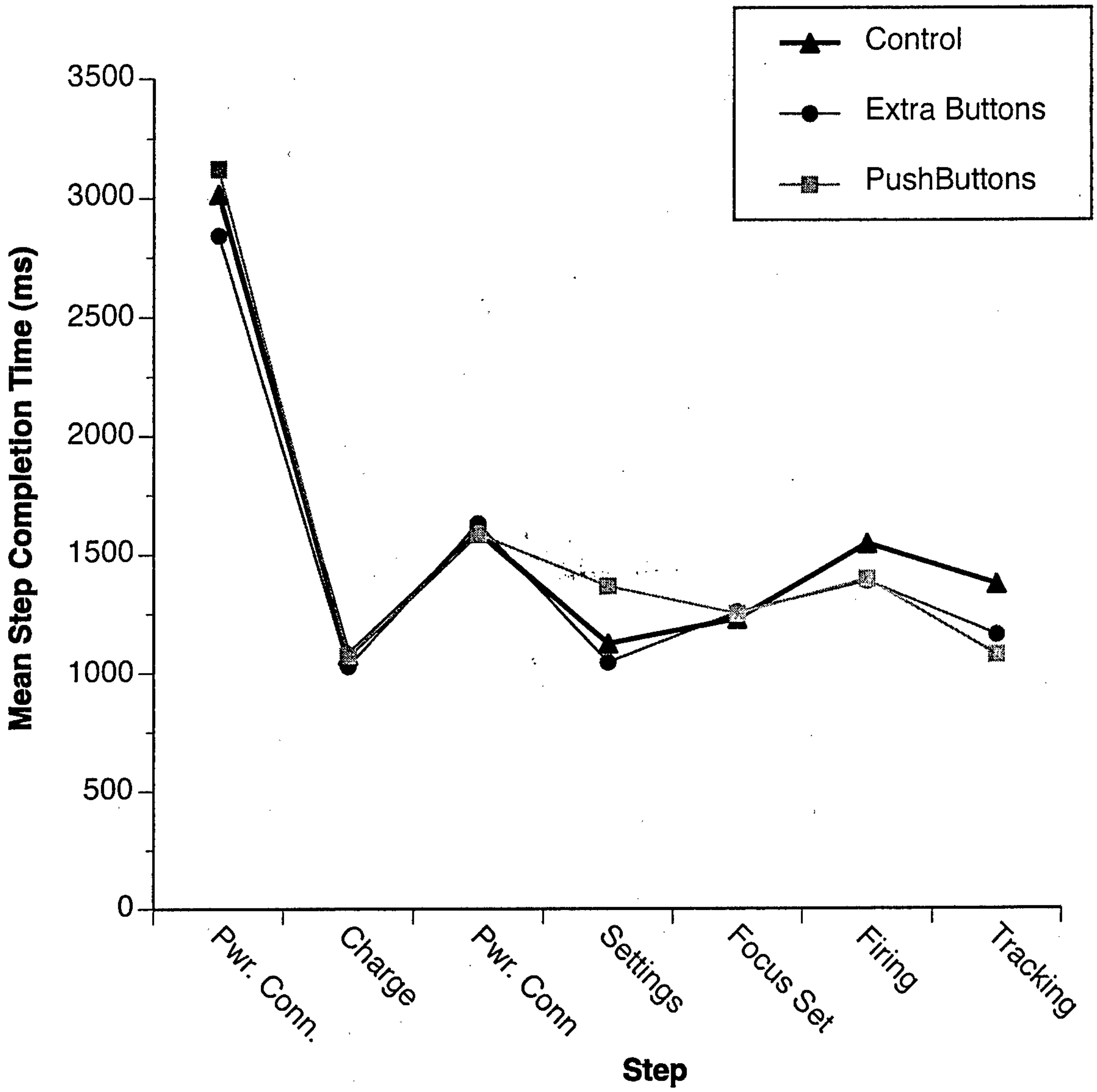


Figure 7. Step completion time by button type condition. See body text for full explanation.

3.2 Computational Cognitive Modeling

Initial models of these tasks were constructed using the ACT-R cognitive architecture (Anderson, et al., 2004). These models are described in more detail in Byrne, Maurier, Fick, and Chung (2004), which is presented in Appendix A and so the detail presented here will be limited. While the modeling efforts have not proceeded as rapidly as hoped, due primarily to the preponderance of empirical results which were unforeseen, the modeling work has nonetheless generated important insights. First, it should be noted that the models have achieved "ballpark" accuracy at reproducing the human execution time data. However, the models do not yet err with human-like frequency. The most important result achieved to date with the modeling work concerns the relative importance of cognitive control structures (e.g., goal management) vs. visual search control. GOMS-style accounts have traditionally emphasized the former, but our modeling work clearly demonstrates that the latter can be at least as important for modeling routine procedures, the intended domain for GOMS models.

The other important insight which came from the modeling work was the inspiration for the empirical investigation into surface features of the buttons described in the previous section. This came directly from examination of the model, which itself does not make use of state information and is only slightly affected by extraneous controls.

3.3 Empirical results: Postcompletion Error

One important class of error which can occur in routine procedural tasks are postcompletion errors. These are omissions of some step or subgoal of the procedure which has the property that it must be executed after the main goal for the task has been satisfied. Standard examples include leaving one's bankcard in an automated teller machine or leaving the original document on the glass of a photocopier or flatbed scanner. While multiple authors had commented about postcompletion errors (e.g., Polson, et al., 1992; Young, et al., 1989), the first laboratory demonstration of this error was Byrne and Bovair (1997). Byrne and Bovair suggested that this error is so robust under conditions of high working memory load that the only viable solution is to design them out. However, Altmann and Trafton (2002) suggested that these errors could be mitigated with appropriate cueing.

Thus, a number of cueing manipulations were investigated and an effective cue was indeed found: a just-in-time, highly visually salient, highly specific cue (red and yellow blinking arrows pointing at the control to be acted upon). This cue did not merely reduce the incidence of postcompletion error; it entirely eliminated the error. See Figure 8 for results. (This work is described in great detail in Chung and Byrne, 2004, which appears in Appendix B.)

Since another highly salient visual cue (a mode indicator) failed to act as a mitigator, further experiments examined the relevant properties of the cue. Weakening the cue by presenting it prior to the appropriate time rendered the cue ineffective. Using a cue which was less specific did mitigate the error somewhat, but not to the same degree. Reducing cue salience (constant red, no blinking), but retaining specificity and appropriate timing, yielded a highly effective cue. See Figure 8 for one of these results.

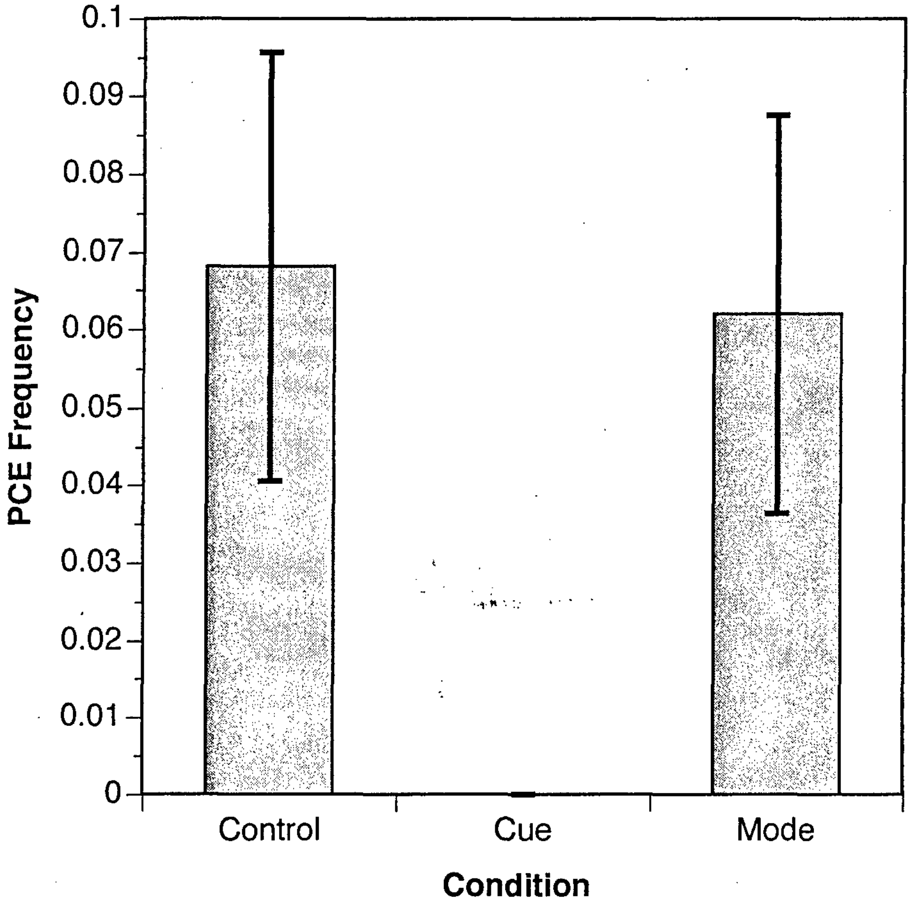


Figure 8. Postcompletion error frequency by mitigation condition. Cue = just-in-time blinking red arrows, Mode = highly-salient mode indicator. Error bars indicate one standard error of the mean.

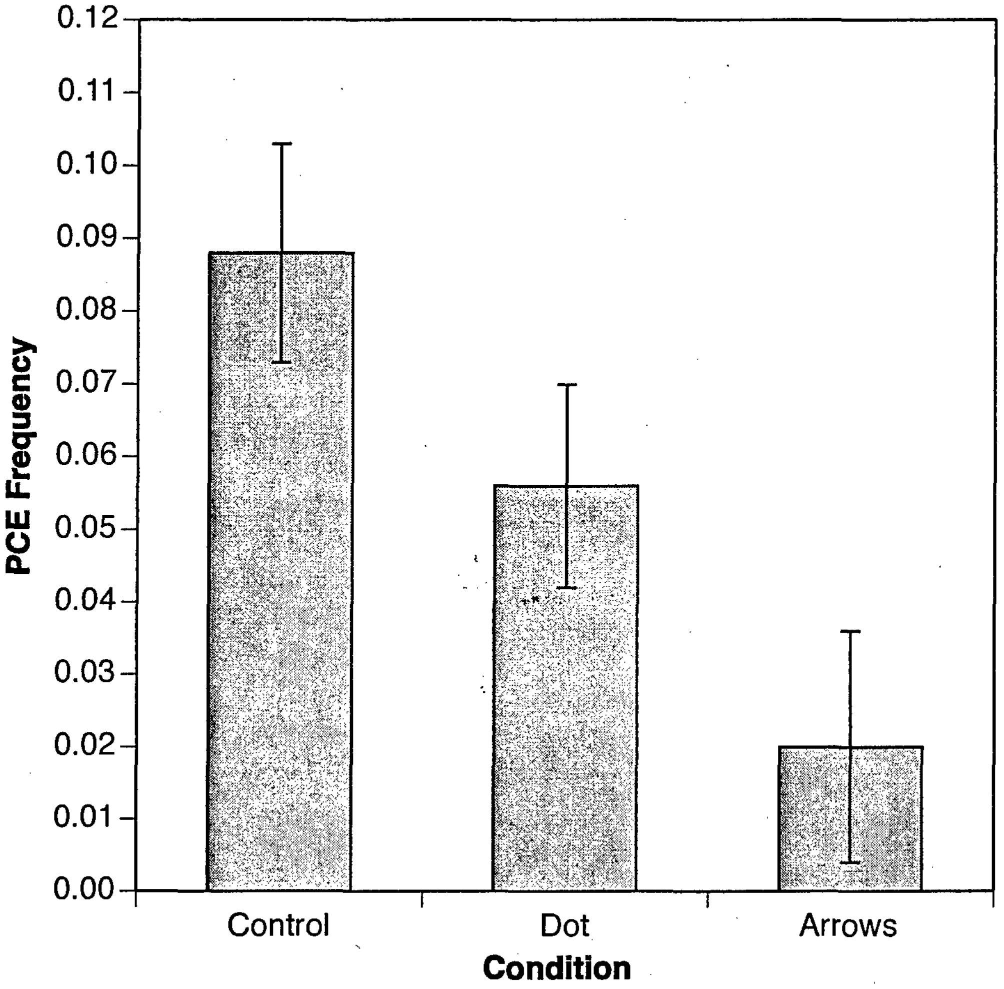


Figure 9. Postcompletion error frequency vs. cue type. Dot = just-in-time blinking dot (salient, nonspecific), Arrows = just in time non-blinking arrows (less salient, specific). Error bars indicate one standard error of the mean.

4. Other Considerations

4.1 Theoretical Perspectives

Finally, another activity undertaken during the period of support was the further consideration of how computational cognitive models can be used to both drive basic research as well as inform real-world applications. This has included work on integrative approaches, surveys of the modeling literature, and position pieces on the role of modeling.

4.2 Technology Transfer

In the near term, technology transfer for the funded work under this grant is happening primarily as a function of an SBIR given by ONR to D. N. American, on which the PI for this grant has been hired as a consultant. Insights gained from the funded research can be brought to bear through this conduit more rapidly than through traditional journal publication channels. This transfer will includes both computational modeling methodology as well as insights into modeling human error. Additionally, data collected in the ONR-funded effort may also be shared with D. N. American to help accelerate their Navy-oriented SBIR work.

In the longer term, technology transfer should happen through multiple channels. One is the publication and presentation of empirical and modeling results in conferences and journals, making them widely available. This has obviously begun, but further publications are in progress. Working with the ACT-R architecture enables another more subtle form of technology transfer. Because the PI is one of the system architects, any enhancements made to the architecture as a result of this research will be propagated to a larger community of researchers, namely the ACT-R modeling community, which includes researchers at various Navy sites as well as others in the DoD community.

5. References

5.1 Publications Supported by N00014-03-1-0094

Byrne, M. D., Fick, C. S., Chung, P. H., & Davis, E. (in preparation). Systematic errors in the execution of isomorphic routine procedures. Manuscript to be submitted to *Journal of Experimental Psychology: Applied.*

Byrne, M. D. (in press). Local theories vs. comprehensive architectures: The cognitive science jigsaw puzzle. To appear in W. Gray (Ed.) *Integrated Models of Cognitive Systems* (title tentative). New York: Oxford University Press.

Chung, P. H., & Byrne, M. D. (accepted 10/27/2005). Cue effectiveness in mitigating postcompletion errors in a routine procedural task. Manuscript conditionally accepted to *International Journal of Human-Computer Studies*.

Byrne, M. D., & Davis, E. M. (in press). Task structure and postcompletion error in the execution of a routine procedure. To appear in *Human Factors*.

Byrne, M. D. (2005). Cognitive architectures in HCI: Present work and future directions. In *Proceedings of Human-Computer International 2005*. Mahwah, NJ: Erlbaum..

Fotta, M. E., Byrne, M. D., & Luther, M. S. (2005). Developing a human error modeling architecture (HEMA). In *Proceedings of Human-Computer International 2005*. Mahwah, NJ: Erlbaum.

Byrne, M. D., Maurier, D., Fick, C. S., & Chung, P. H. (2004). Routine procedural isomorphs and cognitive control structures. In C. D. Schunn, M. C. Lovett, C. Lebiere & P. Munro (Eds.), *Proceedings of the Sixth International Conference on Cognitive Modeling* (pp. 52-57). Mahwah, NJ: Erlbaum.

Chung, P. H., & Byrne, M. D. (2004). Visual cues to reduce errors in a routine procedural task. *Proceedings of the Twenty-Sixth Annual Conference of the Cognitive Science Society.* Austin, TX: Cognitive Science Society.

Byrne, M. D., Chung, P. H., Fick, C., & Maurier, D. (2004). Mitigating errors in the execution of routine procedures. Poster presented at the 2004 meeting of the Human Factors and Ergonomics Society Houston Chapter, Houston, TX, April 2004.

Byrne, M. D., & Gray, W. D. (2003). Returning human factors to an engineering discipline: Expanding the science base through a new generation of quantitative methods—preface to the special section. *Human Factors, 45,* 1–4.

Byrne, M. D. (2003a). A mechanism-based framework for predicting routine procedural errors. In R. Alterman & D. Kirsh (Eds.) *Proceedings of the Twenty-Fifth Annual Conference of the Cognitive Science Society.* Austin, TX: Cognitive Science Society.

Byrne, M. D. (2003). Cognitive architecture. In J. A. Jacko & A. Sears (Eds.), *The human-computer interaction handbook: Fundamentals, evolving technologies and emerging applications* (pp. 97–117). Mahwah, NJ: Lawrence Erlbaum.

5.2 Other References Cited

Altmann, E. M., & Trafton, J. G. (2002). Memory for goals: An activation-based model. *Cognitive Science, 26,* 39–83.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Quin, Y. (2004). An integrated theory of the mind. *Psychological Review, 111*, 1036-1060.

Byrne, M. D., & Bovair, S. (1997). A working memory model of a common procedural error. *Cognitive Science, 21,* 31–61.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Casey, S. (1998). *Set phasers on stun* (2 ed.). Santa Barbara, CA: Aegean.

John, B. E., & Kieras, D. E. (1996). The GOMS family of user interface analysis techniques: Comparison and contrast. *ACM Transactions on Computer-Human Interaction, 3,* 320-351.

Ohlsson, S. (1996). Learning from performance errors. *Psychological Review, 103,* 241–262.

Polson, P. G., Lewis, C., Reiman, J., & Wharton, C. (1992). Cognitive walkthroughs: A method for theory-based evaluation of user interfaces. *International Journal of Man-machine Studies, 36,* 741-773.

Rasmussen, J. (1987). The definition of human error and a taxonomy for technical system design. In K. D. J. Rasmussen, & J. Leplat (Ed.), *New Technology and Human Error (pp.* 53–62). Chichester: John Wiley & Sons.

Reason, J. T. (1990). *Human error.* New York: Cambridge University Press.

Senders, J. W., & Moray, N. P. (1991). *Human error: Cause, prediction, and reduction.* Hillsdale, NJ: Lawrence Erlbaum.

Young, R. M., Barnard, P., Simon, T., & Whittington, J. (1989). How would your favorite user model cope with these scenarios? *ACM SIGCHI Bulletin, 20,* 51-55.

Appendix A. Byrne, Maurier, Fick, & Chung (2004)

Routine Procedural Isomorphs and Cognitive Control Structures

Michael D. Byrne, David Maurier, Chris S. Fick, Philip H. Chung
{byrne, dmaurier, cfick, pchung}@rice.edu
Department of Psychology
Rice University, MS-25
Houston, TX 77005

Abstract

A major domain of inquiry in human-computer interaction is the execution of routine procedures. We have collected extensive data on human execution of two procedures which are structurally isomorphic, but not visually isomorphic. Extant control approaches (e.g., GOMS) predicts they should have the same execution time and error rate profiles, which they do not. We present a series of ACT-R models which demonstrate that control of visual search is likely a key component in modeling similar domains.

Introduction

Every day, people execute countless procedures which are more or less routine. Many of these are uninteresting, but many of these occur in contexts such as emergency rooms and command-and-control centers where failures of speed or correctness can have serious consequences. Thus, understanding how humans execute routine procedures is critical in at least some domains. Card, Moran, and Newell (1983) and John and Kieras (1996) define a routine cognitive skill as one where the person executing the skill has the correct knowledge of how to perform the task and simply needs to execute that knowledge. Roughly speaking, that can be thought of as the point where people are no longer problem solving, but rather applying proceduralized knowledge to a relatively familiar task.

This level of skill has been the focus of attention for an entire family (the GOMS family; John & Kieras, 1996) of techniques for analysis and execution time prediction. This is largely due to the fact that such a wide array of situations fall under this classification, from occasional but not infrequent programming of VCRs to situations involving highly-motivated people in safety-critical situations, such as commercial pilots and medical professionals. As noted, GOMS, which stands for goals, operators, methods, and selection rules, is one of the primary techniques for predicting human performance under these conditions, and the empirical success of GOMS is well-documented (again, see John & Kieras, 1996). A typical GOMS analysis is based on a hierarchical goal decomposition and then a listing of the primitive operators needed to carry out the lowest-level goals. Thus, GOMS analyses are highly sensitive to the goal-based task structure and the number of primitive operations required.

What such an analysis predicts is that two tasks with the same goal/method/operator structure should produce identical performance. While this may be true in a great many cases, it is not universally true. We will present data from two tasks which would yield equivalent GOMS models (which we term "GOMS-isomorphic") but produce significantly different profiles in terms of the time of execution for each step, as well as the error rates at each step. This is not intended as a criticism of the GOMS modeling approach, but rather as the identification of an opportunity for improvement.

This presentation will focus on performance in a series of laboratory experiments in which participants were trained on a number of relatively simple computer-based tasks and then in a subsequent session, returned to perform those tasks along with a concurrent memory-loading task. This paradigm is essentially the same as that used in Byrne and Boviar (1997), which focused on a particular type of procedural error, the postcompletion error. This line of research is primarily concerned with errors made in the task, but to fully understand the errors made, we felt it would first be necessary to understand the cognitive control structures which would produce execution times similar to those we found in the lab. In order to understand these experiments, a relatively thorough understanding of the tasks is required.

The Tasks

Common Procedures

The two tasks under examination were both set in a fictional Star Trek setting to encourage engagement of the undergraduate participants. Participants came in for two sessions spaced roughly one week apart. The first session was training, in which participants were given a description for each task and a manual, walked through the task once with the manual in hand, and then had to repeat each task until they performed it without error three times. In the second session, participants performed the tasks on which they were trained in the first session, along with a concurrent memory-loading task. In this task, they had to monitor a stream of spoken letters which was occasionally interrupted with a beep, after which they responded with the last three letters heard. Participants earned points for correctly executed steps, lost points for errors, received bonus points for rapid performance, and lost points for incorrect answers to the memory probes. High scorers received additional compensation.

While participants were trained on several tasks, not all of which were the same from experiment to experiment, the current research is focused on two tasks, called the Phaser and the Transporter. These two tasks are isomorphic in that they have the same number of steps which were grouped in the training manuals in the same subgoals. The names of those goals, and the names of the buttons and some of the displays and actual controls, however, were different between the two tasks.

The displays for the two tasks appear in Figures 1 and 2 and the list of subgoals and steps appears in Table 1. The main goal in the Phaser task is to destroy the hostile Romulan vessel; the main goal in the Transporter task is to energize it to return some crewmembers to safety. One of the immediately obvious visible differences between the two layouts is that the controls for the Transporter are visually grouped according to subgoal while in the Phaser they are not.

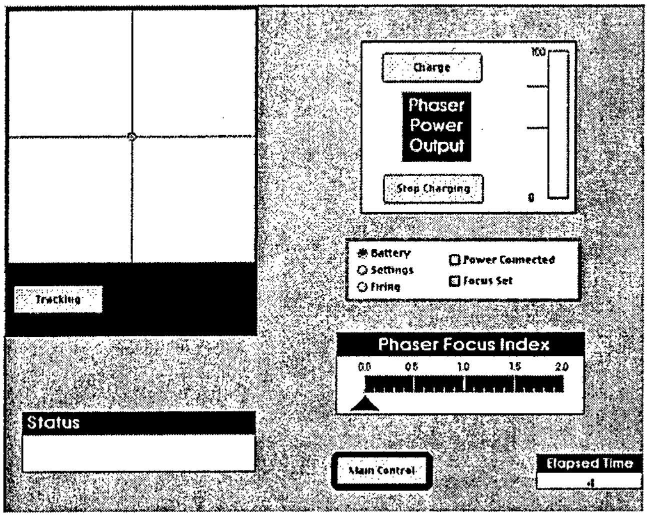


Figure 1. Phaser task display

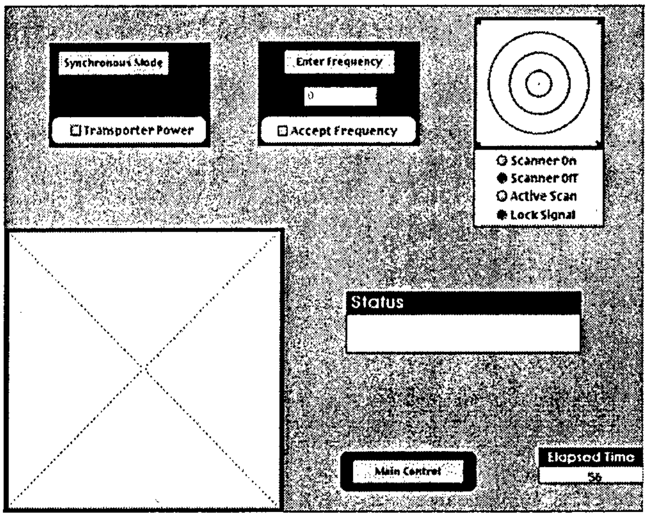


Figure 2. Transporter task display

There are some other important features to note as well. After Step 3 in both tasks, the participants had to wait until the display reached an acceptable state before clicking the next button. Step 6 in both tasks involved, or could involve, multiple actions: multiple drag adjustments to the slider in the case of the Phaser and multiple keystrokes in the case of the Transporter. Step 10 in both procedures involved a somewhat extended tracking task, done with the arrow keys for the Phaser and with the mouse for the Transporter. All of the other steps required the simple clicking of a button.

The exact responses of the display and some of the task structure did differ between the two tasks for Steps 11 and 12 as part of manipulations concerned with postcompletion errors, so those steps were excluded from all present analyses.

The major dependent variables of interest here were step completion time and error frequency Step completion time is measured as the time between clicks. That is, the time for Step 2 in the Phaser is the time elapsed between the click on "Power Connected" and the click on "Charge." For the first step, the start time was the start of the trial. Steps on which errors were made were excluded from the time analysis. Times for steps that include other actions (waiting, tracking) were be excluded from the analysis because this other time is difficult to factor out.

Error frequency was also measured. This hinges on the definition of what counts as an error. Each step can be considered a sequential choice (Ohlsson, 1996), so the definition was based on the step, not the action. If any incorrect action was taken at a step, that step counted as an error, regardless of the number of incorrect actions taken. For example, if a participant is at Step 4 in the Phaser task, and they click on the "Settings" button and then the "Firing" button, only one error was recorded because an error was made at that step. Frequency was calculated as the number of error-containing steps divided by the total number of steps executed.

Step #	Phaser	Transporter
First subgoal		
1	Power Connected	Scanner On
2	Charge	Active Scan
3	Stop Charging	Lock Signal
4	Power Connected	Scanner Off
Second subgoal		
5	Settings	Enter Frequency
6	<slider>	<type>
7	Focus Set	Accept Frequency
Third subgoal		
8	Firing	Transporter Power
9	Tracking	Synchronous Mode
10	<tracking task>	<tracking task>
Fourth subgoal		
11	Tracking	Synchronous Mode
12	Main Control	Main Control

Table 1. Steps in the two task isomorphs

Because these tasks are essentially isomorphic, there is no *a priori* reason to necessarily expect different performance on the two tasks (through Step 10), except perhaps slightly longer step completion times for those steps where the mouse has further to go. Nor was assessing such differences the original purpose of the three experiments we will report; those experiments were primarily focused on postcompletion errors in the Phaser task.

Results

While three separate experiments were run, these experiments differed from each other in detail only. Experiment 1 actually included subsequent sessions with a variety of between-subjects manipulations; Experiment 2 added visual cueing at the postcompletion step of the Phaser; Experiment 3 used a cue and a mode indicator to attempt to mitigate postcompletion effects in the Phaser at step 11; the exact point system used in the three experiments differed slightly; etc. However, none of these surface dissimilarities made much difference; the results are nearly identical for all three experiments (inclusion of "experiment" as a between-subjects variable reveals no main effects or interactions involving that variable). Across the three experiments, data from a total of 164 participants were used. Figure 3 shows the results for step completion time for the Phaser and the Transporter, while Figure 4 presents the error frequency. Steps 3, 6, and 10 are excluded from Figure 3 because those steps involve other processes (e.g., tracking) or possibly multiple actions, as described above. Note that both graphs also include the 95% confidence intervals (non-pooled error).

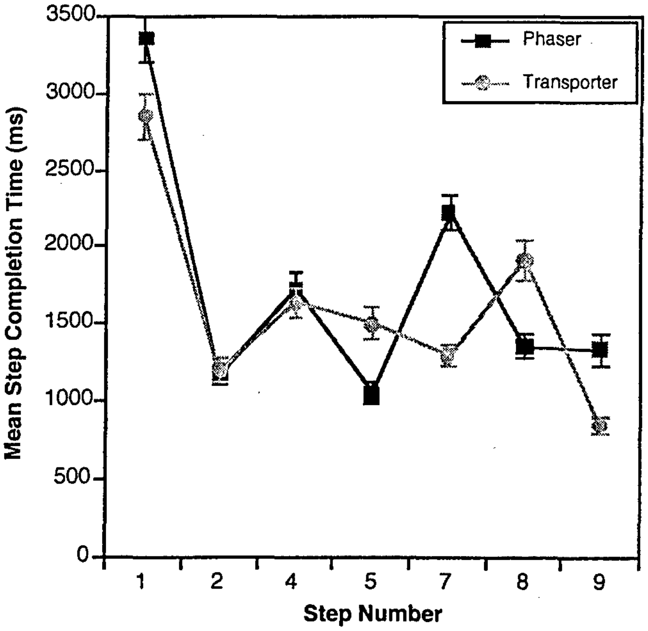


Figure 3. Mean step completion times by task and step

So, while the tasks are isomorphic in terms of subgoals and steps, they produce clearly different step completion time profiles. This runs clearly counter to any account which relies entirely on the GOMS-level structures.

Similarly, if one assumes that the same failure mechanisms are in operation in each task, the two tasks should produce identical error rate profiles as well. This is also obviously not the case. While both tasks share a spike in error rate at step 4 in the procedure (this is, in fact, a postcompletion error), the Phaser shows other spikes at steps 1 and 6 while the Transporter only shows another spike at step 8. Note that these spikes in error rate are not particularly linked to exceptionally large or small step times, either; for example, step 7 in the Phaser is particularly slow, but is not especially error-prone. Step 1 is slow in both tasks, but only markedly error-prone in the Phaser.

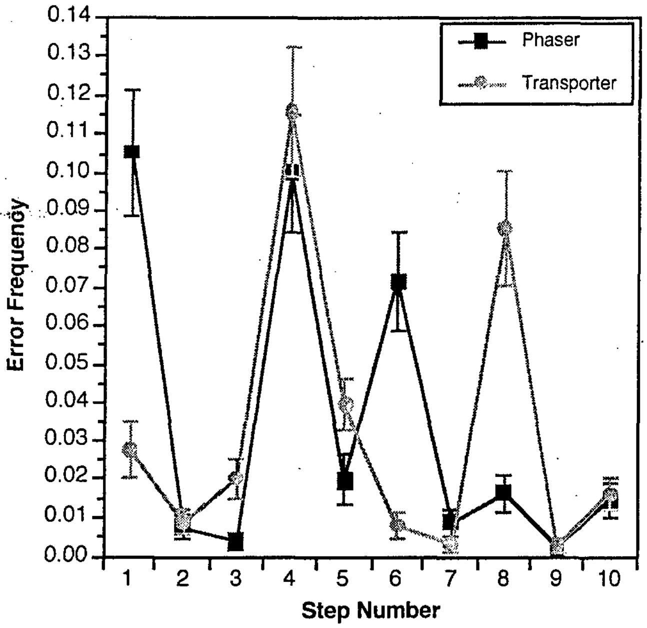


Figure 4. Mean error frequency by task and step

Discussion

These data are obviously problematic for any account which relies solely on the goal-subgoal-method structure for predicting execution time. It is hard to know how a GOMS-style account might accommodate these data. Subjects were probably not at the level of skill where extreme interleaving of cognitive, perceptual, and motor operations is required to model their peformance, thus it is not clear that the CPM variant of GOMS (e.g., Gray, John, & Atwood, 1993) would be appropriate. This is not to say that motor operators are unimportant; there are some differences in going from button to button in terms of pointing time as predicted by Fitts's law, but these differences are relatively small (as will be shown later).

One possibility that seems straightforward is that each of these buttons has to be visually located in order for the mouse to be moved to the button and a click registered. However, there is no single "visual search" operator in GOMS (or ACT-R or Soar for that matter) which would obviously capture the differences here. Each button on the display is at least approximately equal in terms of visual salience; while one might argue that the larger gray pushbuttons are more salient and should thus be found faster, there is little difference between steps 8 and 9 of the Transporter, one of which is a large gray button and the other is simply a labeled checkbox. Furthermore, consider step 7 in the Phaser is markedly slower than step 7 in the Transporter and yet both are simple check boxes with two-word labels. So, if the difference is simply in a "visual search" operator, this operator must itself be driven by something substantially more sophisticated than what is present in a typical GOMS analysis. Furthermore, if the only difference between the two tasks is in their visual search latencies, the source of the differential error spikes remains a mystery.

This obviously raises the question of what kind of control structure could account for the differences between these two tasks? Accounting for the error profiles seems extremely difficult with any model at this point; generative theories of error are in their infancy at best (though that is ultimately our goal, see also Byrne, 2003). Thus, we entered into a modeling exploration with the modest goal of trying to understand what drove the step completion times.

Modeling

We constructed a number of models of this task using ACT-R 5.0 (Anderson, et al., in press). This was done not so much because of a strong commitment to any particular mechanism in ACT-R, but rather because ACT-R contains the full suite of perceptual, motor, and cognitive functionality required for these tasks. It is likely that some version of Soar or EPIC would have served equally well for present purposes but we are much more familiar with ACT-R (and further suspect we will need the subsymbolic mechanisms for future error modeling).

We constructed four models of each task. It was our hope that this way we might "bracket" performance (Kieras & Meyer, 2000; Gray & Boehm-Davis, 2000) and see if the models could provides reasonable predictive bounds. The four models represented a crossing of two dichotomies:

Goal organization. The first dichotomy was whether the model used a hierarchical representation of the goal structure, with intermediate subgoals (e.g., "charge the phaser") or a "flat" goal structure where 12 low-level goals were simply executed in sequence. There is reason to believe that even well-practiced experts do not entirely flatten their goal hierarchies (Kieras, Wood, & Meyer, 1997) and that, in fact, often times fairly slow retrieval-based strategies are appropriate (Altmann & Trafton, 2002). The hierarchical goal strategy is noted with "Hier" in the model label and the flat with "Flat."

Visual search. We implemented two very simplistic visual search strategies here: one in which the location of each button had to be determined through serial visual examination with a tendency to search near the current focus of visual attention (Fleetwood & Byrne, 2003) and one in which the model is assumed to have declarative knowledge of the locations of the buttons which must be retrieved for each button. Various ACT-R models (Ehret, 2002; Anderson, et al. in press) have shown that this kind of learning is a key component of skill development in similar interfaces. The unguided serial search strategy is noted with "DS" (for dumb search) and the alternate with "RL" for "retrieve location."

It should be noted that in ACT-R, these dichotomies may interact. ACT-R's visual system has a memory for which locations (though not explicitly which objects) have been viewed recently, but this memory decays over time (we used 1.5 seconds for this decay time; the models are indeed sensitive to this parameter but in unusual ways which are beyond the scope of this presentation). Thus, additional time spent in traversing the goal hierarchy can result in the loss of this information, which may affect the time course of the serial visual search.

ACT-R also embeds Fitts's law for prediction of mouse movement times. We used ACT-R to calculate the expected movement time between the various buttons to make clear the movement time contribution to the results. We did not compute it for step 1 because the initial location of the cursor was not recorded; informal observation of the participants indicated that many of them moved the mouse around before clicking anyway.

Finally, these models are all stochastic. Time for memory retrievals and perceptual-motor operations in ACT-R can be made noisy and ACT-R chooses randomly between options in various subsystems in cases of ties, so each run of the model is not identical to the last. We present the mean model-generated times for 100 runs of each model.

Model Results

Figure 5 presents the data and the model predictions, as well as the Fitts's Law time, for the Phaser task. Figure 6 presents the same for the Transporter.

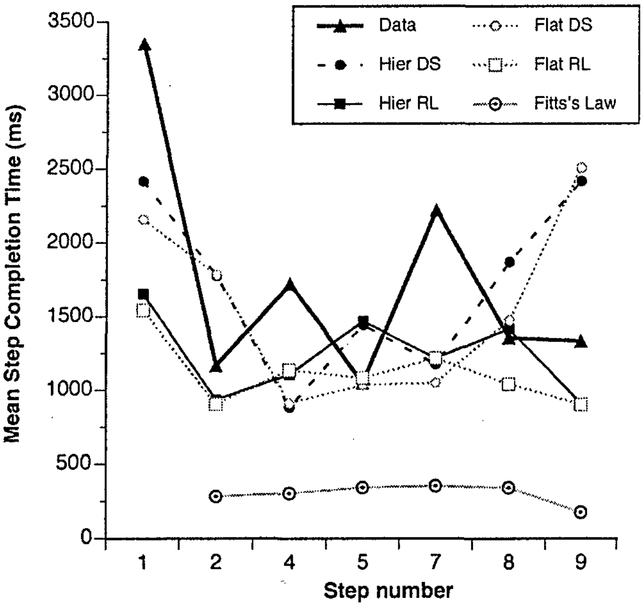


Figure 5. Model and data for the Phaser task

None of the four models provides a particularly good fit; which model is the "best" model by fit metric depends on which metric of fit one uses: by r-squared, the best model is Flat RL at 0.73; by RMSD, the best model is the Hier DS at 640 ms; by mean absolute deviation (MAD) the best model is the Hier RL model at 26%. These are fairly fine distinctions since RMSD ranged from 640–739 ms and MAD ranged from 26–33%. Note that the model variant here which is most similar to a GOMS-style model is the Hier RL model. This model uses hierarchical goal decomposition as per GOMS, and essentially has a fixed time "find-on-screen" operator (the retrieval of the location). This model is generally good, if a bit too fast, for the Transporter, but is a poor model for the Phaser.

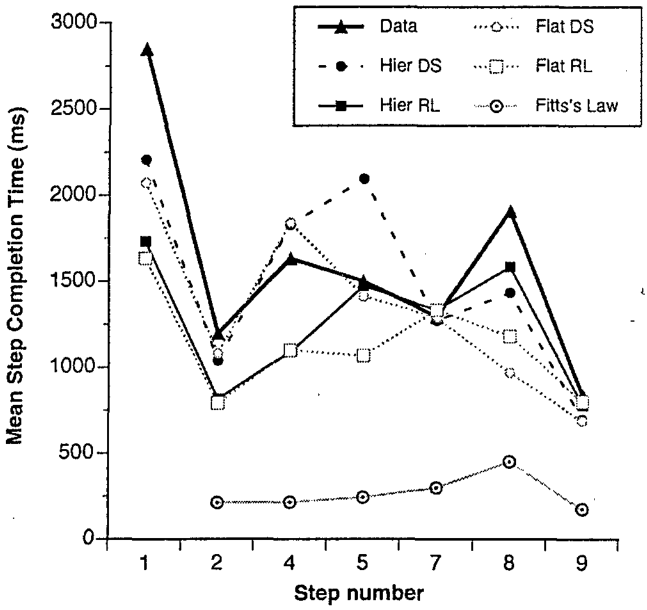


Figure 6. Model and data for the Transporter task

While the models do not provide outstanding levels of fit, they do provide some important insights. First, Fitts's Law alone provides an r-squared of 0.28 for those steps where it is applicable. Obviously time is grossly under-predicted by Fitts's Law—it is hardly surprising that more is going on here than simple motor movement, though it is clearly a contributor.

In general, the ordinal effects one would expect from the basic construction of the models held: the Flat models were faster than the Hier models and the DS models were generally faster than the RL models. Note that in general, the RL models' performance on the two isomorphs was quite similar. This is a reflection of their isomorphic task structure. The DS models, on the other hand, reflect differences between the tasks. This is consistent with the notion that it is the visual aspects of the display—the DS models are sensitive to button location with the RL models are only in the Fitts's law sense—that drives the differences between the two tasks.

However, there were a few cases where the DS and RL models were rougly equivalent, and even one case where the DS models were faster (Phaser step 4, Power Connected). There were some degenerately bad performances by the DS model, notably Phaser step 9, Tracking, and the Hier DS on Transporter step 5, Enter Frequency. Both of these cases involve a visual shift to a location which has a lot of competition from items much closer to the starting attentional focus.

On the other hand, the slower DS resulted in better fit in some instances, namely step 1 for both tasks, and steps 2 and 3 for the Transporter. Step 1 is particularly problematic for all four models; this is a very slow step for both the models and the participants, but more so for the participants. We suspect this is due to some kind of initial orienting or goal construction on the part of the participants which was not well-represented in the model, but may be partially represented by the DS behavior of taking an initial visual survey of the display. This plays into the next insight we gained from these models.

In general, the RL models were slightly better than the DS models. What this suggest to us is that participants in this case are at an intermediate point in their learning of the locations of the objects on the interface. Our next model will likely not start with the locations explicitly encoded in declarative memory but will instead use the strategy of attempting to retrieve them from memory, but this time from the memories created as a by-product of visual searches conducted along the way.

Comparisons between the Flat and Hier models are also revealing. These models differed primarily at steps where either they interacted with the visual search process (Transporter step 5, Enter Frequency is a good example of this) or there was a delay for additional goal traversal (steps 1, 5, and 8. This additional time appears correct for both tasks for steps 1 and 8, but step 5 indicates something else going on. Both Hier models are too slow for step 5 in the Phaser, but the Hier RL model is right on target for step 5 for the Transporter.

Finally, some points were fairly strategy-insensitive. Transporter step 7 (Accept Frequency) was fit equally well by all four models. This is an interesting case for two reasons: [1] the DS visual search strategy will almost always search the correct location first here because of visual proximity to where the model is looking prior to this step, [2] it is the last subgoal within the second goal, and thus not differentially affected by the goal organization, and [3] the completion time for the similar Phaser step is radically different. None of the models captured this deviant time in the Phaser at all.

Discussion

While it may appear that our goal was to somehow falsify or criticize GOMS models, but that was not our intent. Instead, we wanted to explore where and why models based purely on the GOMS-style structure would misfit, not for the purposes of finding fault, but to find opportunities for improvement. One of the primary things GOMS-style models lack is a consideration of the visual task faced by interface users. This was certainly reasonable when most users faced command-line tasks which were indeed primarily cognitive, but the shift to increasingly visual interfaces has raised the importance of systematically addressing the problem of how the visual and cognitive systems are integrated. While this has certainly been a big topic for some cognitive scientists for many years (see Pylyshyn, 1999 and the associated commentary for an excellent discussion), it has not been a prominent theme in computational modeling of human-machine interfaces until fairly recently and in cases where the task is clearly defined as primarily a visual search task (e.g., Fleetwood & Byrne, 2003; Everett & Byrne, 2004; Hornof & Kieras, 1997, 1999) . Our research suggests this may be an important part of routine procedure execution even when visual search may *not* appear to be a dominant factor. Furthermore, it appears that it is neither the case that the most optimistic assumption (users memorize the location of all controls) or the most pessimistic assumption (users search randomly every time) is an appropriate representation of user behavior, at least at this level of skill. This suggests that more research is needed on the integration of cognitive mechanisms such as representing and traversing goal structures with visual-cognitive mechanisms such as search strategies. While we doubt anyone would have denied that this is an important domain in a general sense, we suspect that most researchers in this area would underestimate the impact such considerations might have on execution of routine procedures.

To end on a speculative note, consider the hint provided by Phaser step 5 (settings). In that case, the Hier models are too slow and the Flat models are spot-on, suggesting that the goal traversal performed by the model is not being done by the participants. This might be an indicator that the participants have re-configured their internal representation of the task structure to match the visual structure of the interface! This suggests a possibly important role for the match between the task structure and the visual layout of an interface, something clearly not predicted by extant GOMS-class models.

Acknowledgements

We would like to acknowledge the support of the Office of Naval Research under grant number N00014-03-1-0094 to the first author. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ONR, the U.S. Government, or any other organization.

References

Altmann, E. M., & Trafton, J. G. (2002). Memory for goals: An activation-based model. *Cognitive Science, 26*, 39–83.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Quin, Y. (in press). An integrated theory of the mind. To appear in *Psychological Review.*

Byrne, M. D. (2001). ACT-R/PM and menu selection: Applying a cognitive architecture to HCI. *International Journal of Human-Computer Studies, 55(1)*, 41–84.

Byrne, M. D. (2003). A mechanism-based framework for predicting routine procedural errors. In R. Alterman & D. Kirsh (Eds.) *Proceedings of the Twenty-Fifth Annual Conference of the Cognitive Science Society.* Austin, TX: Cognitive Science Society.

Byrne, M. D., & Bovair, S. (1997). A working memory model of a common procedural error. *Cognitive Science, 21*, 31–61.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Ehret, B. D. (2002). Learning where to look: location learning in graphical user interfaces. In *Human Factors in Computing Systems: Proceedings of CHI 2002* (pp. 211-218). New York: ACM.

Everett, S. P., & Byrne, M. D. (2004). Unintended effects: Varying icon spacing changes users' visual search strategy. *Human Factors in Computing Systems: Proceedings of CHI 2004* (pp. 695–702). New York: ACM.

Fleetwood, M. D. & Byrne, M. D. (2003). Modeling the visual search of displays: A revised ACT-R/PM model of icon search based on eye-tracking and experimental data. In F. Detje, D. Dörner, & H. Schaub (Eds.) *Proceedings of the Fifth International Conference on Cognitive Modeling* (pp. 87–92). Bamberg, Germany: Universitas-Verlag Bamberg.

Gray, W. D., & Boehm-Davis, D. A. (2000). Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. *Journal of Experimental Psychology: Applied, 6*, 322-335.

Gray, W. D., John, B. E., & Atwood, M. E. (1993). Project Ernestine: A validation of GOMS for prediction and explanation of real-world task performance. *Human-Computer Interaction, 8*, 237–309.

Hornof, A. J., & Kieras, D. E. (1997). Cognitive modeling reveals menu search is both random and systematic. In *Human Factors in Computing Systems: Proceedings of CHI 97* (pp. 107–114). New York: ACM Press.

Hornof, A. J., & Kieras, D. E. (1999). Cognitive modeling demonstrates how people use anticipated location knowledge of menu items. In *Proceedings of ACM CHI 99 Conference on Human Factors in Computing Systems* (pp. 410-417). New York: ACM.

John, B. E., & Kieras, D. E. (1996). The GOMS family of user interface analysis techniques: Comparison and contrast. *ACM Transactions on Computer-Human Interaction, 3*, 320-351.

Kieras, D. E., & Meyer, D. E. (2000). The role of cognitive task analysis in the application of predictive models of human performance. In J. M. Schraagen & S. F. Chipman (Eds.), *Cognitive task analysis* (pp. 237-260). Mahwah, NJ: Erlbaum.

Kieras, D. E., Wood, S. D., & Meyer, D. E. (1997). Predictive engineering models based on the EPIC architecture for multimodal high-performance human-computer interaction task. *Transactions on Computer-Human Interaction, 4*(3), 230-275.

Ohlsson, S. (1996). Learning from performance errors. *Psychological Review, 103*, 241–262.

Pylyshyn, Z. (1999). Is vision continuous with cognition? The case of impenetrability of visual perception. *Behavioral & Brain Sciences, 22*(3), 341-423.

Appendix B. Chung & Byrne (accepted)

TITLE: CUE EFFECTIVENESS IN MITIGATING POSTCOMPLETION ERRORS IN A ROUTINE PROCEDURAL TASK

Phillip H. Chung[a] and Michael D. Byrne[a]

[a]Rice University
Psychology Department-MS 25
PO Box 1892
Houston, Texas 77251-1892 USA

Corresponding author:
Michael D. Byrne
+1 713-348-3770 (voice)
+1 713-348-5221 (fax)
byrne@acm.org

Abstract

Postcompletion errors, or errors which occur when some action required after the completion of the main goal is omitted (Byrne & Bovair, 1997), can be found in a variety of everyday procedural tasks. Reducing their frequency by means other than completely redesigning the task structure can be difficult, however, because they are so robust. Finding a successful mitigation strategy may uncover important mechanisms underlying task performance. Two experiments were carried out to test a variety of cues for their ability to mitigate the frequency of postcompletion errors in a computer-based routine procedural task. A just-in-time, specific, and visually salient (blinking) cue entirely eliminated the error, but other cues were ineffective. This is consistent with Altmann and Trafton's (2002) model of memory for goals but somewhat contrary to the predictions of standard usability guidelines. Finally, a computational model was developed in ACT-R to mirror participant performance and validate the theoretical implications of the findings from the two experiments.

Keywords:

postcompletion error; human error; interface design; modeling; ACT-R; cognitive architecture; routine procedural task; error mitigation; error intervention; visual cue; visual attention; visual salience; goal memory

1. Introduction

In one of the most influential works on human error Reason (1990) defined human error as "all those occasions in which a planned sequence of *mental* or *physical* activities fails to achieve its intended outcome, when these failures cannot be attributed to the intervention of some chance agency." This definition does not attempt, however, to explain *why* or *how* such failures occur. Similarly, the intent of most popular error *taxonomies (e.g., Norman, 1988) is not the prediction of errors. For the present perhaps* the best we can look to in terms of error prediction is some probabilistic measure based on an analysis of the task and interface. Such human error identification approaches include Task Analysis for Error Identification (TAFEI) developed by Baber and Stanton (1994) and more statistical approaches, such as Swain and Guttman's (1983) Technique for Human Error Rate Prediction (THERP), both popular in safety-critical settings. All are somewhat lacking, however, in their account for the specific cognitive components and processes leading to erroneous behavior, as well as the conditions surrounding them. Moreover, once a potential error is identified by such a technique, the responsibility falls solely on the evaluator or designer to generate an appropriate solution.

The generation of a theory to support human error prediction in computer-based routine procedural tasks would require significant data above and beyond what currently exists. Such a theory, based on our existing understanding of human cognition, would also be beneficial for the development of specific design solutions. As Rasmussen (1987) once suggested, this type of effort would necessitate a human error "data bank" of sorts. None presently exists, however, as capturing data on low frequency errors in the laboratory is difficult to set up and time consuming (Wood, 2000). Fortunately, the last

century's introduction of automation and computers has established an outstanding arena for this problematic element of human behavior to showcase itself. Subsequently, much effort has been given to retrospectively understand errors (e.g., root cause analysis and error taxonomies), which have occurred in such situations, and generate general design guidelines to reduce them. Nevertheless, according to John and Kieras (1996), "No methodology for predicting when and what errors users will make as a function of *interface design has yet been developed and recognized as satisfactory…even the* theoretical analysis of human error is still in its infancy."

The domain of HCI holds great potential for the study of human error, as software programs provide an easily manipulated stimulus to elicit and capture error. Since computer-based tasks today are primarily visual, simply altering the visual design of the interface can affect patterns of user error (Byrne, Maurier, Fick, & Chung, 2004). From this one may be able to extrapolate principles founded on cognitive and perceptual theory to arm designers and evaluators with solutions for redesign. This would not only improve our *post hoc* evaluations but also enable the design of safer computer interfaces. Application areas to benefit abound, from aerospace to medicine. In this paper two experiments are reported as well as a novel approach to studying error using the ACT-R cognitive architecture (Anderson, Bothell, Byrne, Douglass, Lebiere, & Quin, 2004). This work provides an interesting demonstration of the difficult task of human error identification from a cognitive perspective, which existing methods of cognitive task representation and error identification do not adequately manage.

Current theory in psychology suggests that the same processes producing successful human performance can also be looked to as the source of error (Baars, 1992;

Reason, 1990). Reason (1990) describes this in terms of a "cognitive balance sheet," where correct performance and systematic errors are placed on opposing sides. For example, the emergence of automatic behavior through delegation of control to lower level processes introduces the opportunity for behavioral "slips" to arise. Hence, in order to understand how errors occur, it is necessary to consider the cognitive mechanisms that govern correct human behavior.

Rasmussen's (1987) "Skill-Rule-Knowledge" (SRK) description of task performance provides a general framework of cognitive control mechanisms, which can be used to describe how errors occur. It identifies three separate levels of cognitive control displayed during task performance: skill-based, rule-based, and knowledge-based. Each corresponds to a different degree of familiarity with the task and environment, with knowledge-based behavior representing the least degree of control and familiarity and skill-based the highest. With experience a person proceeds sequentially through the three stages of the model, moving from lowest (knowledge-based) to highest (skill-based). At the rule-based level, rules for behavior are selected using selection criteria (listed below) based on the mental model the operator has constructed in their mind about a system.

1. *Match* to salient features of the environment or internally generated messages.
2. *Strength* or the number of times a rule has performed successfully in the past.
3. *Specificity* to which a rule describes the current situation.
4. *Support* or the degree of compatibility a rule has with currently active information.

Failure modes stem from either the application of bad rules or misapplication of good rules due to incorrect rule selection. These rules may be active simultaneously, with several competing for instantiation. These also control the occurrence of errors at the rule-based level, in keeping with Reason's (1990) idea of a "cognitive balance sheet."

1.1 Postcompletion Error

Noting the general lack of specificity in the existing theories of human error, Byrne and Bovair (1997) moved to develop a computational theory for one widely cited (e.g., Rasmussen, 1982; Young, Barnard, Simon, & Whittington, 1989) omission error, postcompletion error. Postcompletion errors can be roughly defined as errors that occur when the task structure demands "that some action…is required after the main goal of the task…has been satisfied or completed," (Byrne & Bovair, 1997, p. 32). Some commonplace examples include forgetting to remove the original after making a photocopy, leaving a card in the ATM after withdrawing cash, and failing to replace the gas cap after filling up a car. With this particular class of error, the actor possesses the correct knowledge necessary to execute the task, which is usually performed correctly yet still generates systematic errors.

Even for operators highly familiar with the task, the isolation of a postcompletion step within the task structure makes omissions there likely (Figure 1). This is particularly true when the actor is further affected by external factors such as a working memory load and/or fatigue, as well as internal human tendencies such as hillclimbing (Gray, 2000; Polson & Lewis, 1990). Byrne and Bovair (1997) hypothesized that these errors were due to excessive working memory load leading to goal loss, or an omission of a step from the task at hand. Since with postcompletion errors the actor omits a specific subgoal rather than forgetting what to do altogether (the overlying main task goal), the source of the error was thought to more likely be working memory than long-term memory. A more recent study by Reason (2002) examined the photocopy example in detail, finding postcompletion errors to be the most common type of omission in that task.

<< Figure 1 >>

Three high-level explanatory observations were provided:

1. The emergence of the last copy generates a strong but *false completion signal* since the main goal of copying is achieved before all necessary steps (subgoals) are complete.
2. The proximity of this false signal to the end of the main task allows for the *attention to be increasingly diverted* to the subsequent task.
3. The emergence of the last copy indicates that it is no longer necessary to put in another original leaving it *functionally isolated.*

Because commission of this type of error is reliable under high working memory load, Byrne and Bovair (1997) suggest the only absolute solution is to design it out. It is commonly recommended to rearrange the task so that the user is forced to complete the otherwise potentially omitted step in order to achieve the main task goal. Automated teller machines initially faced the same sort of problem as the auditory and visual reminders implemented in early models failed to reliably remind customers to withdraw their card. As a result, many ATMs today now feature a forcing function that prevents the user from proceeding with a transaction before the card is withdrawn. For the gas cap example, newer cars now have caps that are somehow physically attached to the car.

1.2. Hierarchical Control Structures and Goal Management

Many of the assumptions behind the theory of postcompletion error reside on the concept of hierarchical control structures and their retention by skilled operators. In previous studies by Byrne and Bovair (1997) and Serig (2001), participants reliably generated errors at the postcompletion steps both within subtasks as well as within the larger task, in keeping with the idea of a hierarchical task structure. Cognitive modeling work by Kieras, Wood, and Meyer (1997) has also provided strong evidence to suggest

that even well practiced experts, such as telephone assistance operators, do not abandon such task hierarchies. Altmann and Trafton (1999) propose that this ability to break down complex tasks and problems into hierarchies and subgoals, "may be to complex cognition what the opposable thumb is to complex action."

Traditionally, these types of goal-based processing strategies have relied solely on a "task-goal" stack that essentially predicts perfect memory for old goals. However, Altmann and Trafton's (2002) activation-based model of memory for goals (MAGS) offers an alternative account to this approach that provides a more straightforward account for the types of errors found in human behavior. In essence memory and the environment (i.e., dual-space, Rieman & Young, 1996; internal and external representations, Zhang & Norman, 1994) are substituted for a goal stack, and task goals are.considered as ordinary memory elements with encoding and retrieval processes that must overcome noise and decay. Retrieval cues from the environment dictate the reactivation of suspended goals with perceptual heuristics acting as a substitute for the stack-native last in, first out rule. This model makes several predictions about postcompletion errors and the characteristics of a successful cue:

1. Any *salient* cue (e.g., a loud beep) should be sufficient to prime a postcompletion action.
2. It should *not* be necessary to put the postcompletion action on the critical path.
3. Reminders at the start will *not* help a PCE at the end because they are masked by other goals.
4. Just-in-time priming from environmental cues are the *only* reliable reminder.

1.3.Task Modeling and Human Error Identification

Hierarchical control structures and goal management are major components of popular cognitive task modeling approaches. Hierarchical Task Analysis (HTA), initially developed in the 1960s to aid the investigation of complex non-repetitive tasks, forms the basis of human error identification methods such as Baber and Stanton's TAFEI method (1994). By breaking down the task goal structure hierarchically, the human side of the interaction can be modeled in conjunction with state space diagrams detailing the behavior of the artifact. Similarly, GOMS (Goals Operators Methods and Selection Rules; John & Kieras, 1996) offers a means to represent knowledge required by humans in computer-based tasks for correct performance. By representing this information in such a manner, it becomes possible to analyze the dynamics of the interaction, even to the point of identifying potential errors, as has Wood (2000). Using usability guidelines, Wood (2000) was able to evaluate a GOMS model representing interaction with an e-commerce website to guide development of channels for error recovery. For example, in accordance with a heuristic to reduce working memory load, the GOMS model was used to identify procedures that were particularly taxing on working memory. The design of the system was then iteratively reworked to eliminate these particular procedures.

In both of these and the several other extant methods of human error identification such as Cognitive Reliability and Analysis Method (CREAM; Hollnagel, 1998) and THERP (Swain & Guttman, 1983), correct performance must first be modeled in some form before the analyst can then proceed to "predict" or identify potential errors in the task procedure. This falls in line with the notion that correct performance and human error are closely tied. Nonetheless, two major weaknesses afflict these techniques (Stanton, 2004), the first of which is their weak or nonexistent account for the external environment including stress and noise. THERP, most notably, has been criticized for its reliance on fixed error probabilities and lack of account for varying levels of stress. Even

representation of the system under analysis is limited to simple space-state diagrams at best (e.g., Baber, 1996). As a result, existing methods are rather ill-suited to predict error in highly perceptual and dynamic tasks such as flying a plane or driving a car. The second major drawback to these methods is that they rely heavily on the skill and/or judgment of the analyst. This can lead to both inter- and intra-analyst reliability problems for obvious reasons. Despite their drawbacks, however, there are good reasons for why techniques such as THERP continue to thrive for certain applications such as power plants (Kirwan, 1992). Our intention is not to belittle their significance but rather seek improvement to deal with more highly perceptual and dynamic computer-based tasks. One direction to take is to study the visual components of human error in computer-based routine procedural tasks with the use of visual cues.

1.4. Perceptual Errors and "Salience"

Both Reason's (2002) and Altmann and Trafton's (1999) recommendations for handling postcompletion errors rely on the stipulation that a mitigating cue or reminder is salient or conspicuous. The MAGS theory (Altmann & Trafton, 2002) stipulates that some earlier cue (internal or external) be associatively linked to a subsequent target, such as the postcompletion action. In fact, they take the tendency of skilled users to generate correct behavior most of the time as "evidence of deliberate cognitive operations undertaken to meet the priming constraint—to ensure the existence of an associative link to the postcompletion action, and to ensure attention to the right cue at the right time," (p. 64, Altmann & Trafton, 2002). This is a good working hypothesis as to what occurs when *correct* behavior is displayed.

Nevertheless, in instances where a postcompletion error does arise, it is left to establish if the *incorrect* behavior can be traced to the stage of perception (failure to see the cue), attention (failure to attend to the cue), or goal association and retrieval (retrieving the incorrect goal or failing to retrieve a goal). This requires one to determine if the cue is conspicuous or salient, particularly under the external conditions of the task. In road accidents, for example, failures at the stage of attention are most common, as drivers fail to attend to a plainly visible object such as a road sign (Green & Senders, 2004). Results from the following experiments demonstrate the difficulty of tracing errors in these instances, an undertaking that extant error identification methods leave to the designer or analyst to overcome with their "expert judgment."

2. Experiment 1

While designers may opt to place the hanging postcompletion action "on the critical path" to reduce or eliminate their omission, such as with many ATMs, this is often impossible or too expensive given the existing system. It is well known that the selection or noting of visual cues or features is automatic (e.g., Treisman, 1986). Automatic processing of novel peripheral cues regardless of whether or not they are informative has been well documented in the literature (e.g., Jonides, 1981; Remington, Johnston, & Yantis, 1992). Evidence suggests that such cues are even more powerful when people are directed to look for a specific feature (e.g., Most et al., 2000). Applied research has similarly demonstrated that even the addition of a simple visual cue such as an orange dot can bring about changes in how people interact with physical objects such as doors (Wallace & Huffman, 1990). In light of these theories and experimental evidence, it seems viable to reduce postcompletion errors or omissions in an interactive task by simply cueing or priming a suspended goal. If a necessary condition for slips is attentional capture of some form, external reminders or environmental cues similarly exploiting attentional capture should be able to reactivate the suspended subgoal for the postcompletion action at the appropriate time. Similarly functioning cognitive aids are frequently used in industrial settings and aviation to reduce human error. By prompting the actor to make sure that all steps have been completed in the task, they fundamentally augment the limited capacity of working memory that is the root cause of many errors. Furthermore, many existing everyday devices and computer applications have indicator lights, beeps, alarms, etc. that act as reminders, although their efficacy may sometimes be questionable.

Another possible means of reducing errors over time is to introduce a downstream error "cost" for incorrect performance in the task. In this case making an error on a given trial leads to a cost in time or effort further down in the task sequence. For instance, forgetting to save ones work before shutting down a program would require one to go back and redo the work to successfully finalize the task. In theory, the cost of having to do the work twice should provide incentive for a change in behavior. If feedback is immediate and obvious, then it may be justifiable to predict that implementing such a cost into a computer-based task would promote execution of the correct behavior by the operator in subsequent interactions.

2.1. Predictions

Detailed analyses of such simple and practical countermeasures to error and their corresponding human response will facilitate better comprehension of the mechanisms behind human error within human-computer systems and what can realistically be done to offset them. Using modified versions of the computer-based tasks employed by Serig (2001) and Byrne and Bovair (1997), the objective of Experiment 1 was foremost to study the effects of: (1) a simple automated visual cue and (2) a downstream error cost on postcompletion error. Postcompletion errors are well accessible for study, having been shown to be robust and reproducible in previous inquiries and observations from real-world settings (e.g., Byrne & Bovair, 1997; Reason, 2002; Serig, 2001).

The idea of humans as "cognitive misers," their further tendency to hillclimb, evidence for display-based difference reduction (Gray, 2000), and the predictions of MAGS (Altmann & Trafton, 1999) all suggest that humans will use whatever means is made available to them and cut corners to reduce cognitive work. Thus, it was predicted that participants would exploit the simple visual cue as a reminder to complete the postcompletion step. Since participants were given extensive training on the system and the cue, they should have formulated an adequate mental model of the system allowing them to pay attention to the cue at the appropriate time. The cue adhered to recommendations by Reason (2002) to be conspicuous (at the critical time) and contiguous (proximal), research by Yantis and Jonides (1988) showing onsets to capture attention better than color or intensity, predictions by Altmann and Trafton (1999) regarding just-in time priming from environmental cues, and real-world prominence on existing devices and applications.

It was also hypothesized that feedback in the downstream error "cost" condition would help participants remember to complete the postcompletion step on subsequent trials. With the target task this condition was instantiated by leaving the console at the

postcompletion step if it was omitted on a previous trial. In such a case, when the participant returned to the system, it failed to respond until the formerly omitted postcompletion step was first executed. Feedback in the form of cost in time and points incurred by having to reorient oneself with the awkward state of the system and remember to complete the postcompletion step was expected to cause participants to devote increased attention at the postcompletion step on subsequent trials. This would potentially be driven by the desire to reduce overall effort, as predicted by the cognitive miser account.

2.2.Method

2.2.1. Participants

81 (36 male, 45 female) undergraduate and graduate students from Rice University aged 18-30 participated for course credit in a psychology course and/or prizes. Amazon gift certificates were awarded to the top three finishers ($25, $15, and $10).

2.2.2. Materials

The materials for this experiment consisted of a paper instruction manual for each of the three tasks (Navigation, Transporter, and Tactical), Apple iMac computers running the Bridge Officer Qualification application written in Macintosh Common Lisp, and Sony MDR-201 headphones. The manuals were thorough in detail and offered illustrations of the interface and diagrams for each step. Organization of the instructions was hierarchical (see Figure 2), in accordance with the idea of hierarchical control structures. Step-by-step summaries at the end of each manual provided a complete picture of the task for review.

<< Figure 2 >>

2.2.3. Design

This experiment used a two-factor within (trial at testing) and between (task assignment) participants design. Task assignment consisted of four conditions: control (no intervention version of the Tactical task), cued (cued version of the Tactical task), mode error (mode error version of the Tactical task), and a combined cued/mode error condition. Participants were randomly assigned to one of these four conditions. The primary dependent variable was the frequency of postcompletion errors made during the Tactical task (out of the total number of opportunities). Completion time at the postcompletion step was also a measure of interest.

2.2.4. Procedure

Participants were run in two sessions, spaced one week apart. The first session served as a training session using written documentation for each of the tasks: Navigation, Tactical, and Transporter. Order of training on the three bridge station tasks was randomized for every participant, as was group assignment. Only the Tactical task (see Table 1) contributed to the measure of postcompletion error. Once participants successfully completed the training trial and logged three subsequent error-free trials, they were allowed to move on to the next task. Errors resulted in warning beeps and

messages, ejected the operator to the main control, and restarted the task. This prevented participants from completing training without having gone through each of the tasks at least four times with all steps done correctly and completely. When training was complete for all three tasks, they were reminded that they would be competing for prizes in one week.

The second session consisted of the test trials for the three bridge station tasks. Participants completed thirteen trials of each task in random order, for a total of thirty-nine trials for the test day. During the second session, the experiment program emitted warning beeps on error commission to warn individuals but did not eject them to the main control as in training. Moreover, warning messages and reminders were removed and trials were continued until the task goal was met.

The concurrent working memory letter task was also introduced on the day of testing. As in the studies by Serig (2001) and Byrne and Bovair (1997), its function was to increase working memory load during task performance. Participants were presented with auditory stimuli in the form of randomly ordered letters spoken through the headphones at a rate of one letter every three seconds. A tone was presented randomly at intervals ranging from nine to forty-five seconds upon which the participants were directed to recall the last three letters in order and type them into the text box that appeared on the screen. This was the same for all four conditions.

Participants were encouraged to work both accurately and quickly by means of a scoring system, an onscreen timer, and prizes. The scoring system incremented twenty-five points for each correctly executed step and decremented fifty points for each incorrect. Up to 100 points were awarded for task completion within a set time. For every incorrect working memory recall trial, the score was decremented 200 points. No points were accumulated for successfully completing a recall trial. The large weight placed on the recall task was due to an observation made during a previous experiment (Serig, 2001) of participants tending to neglect the working memory task. At the end of each trial, participants were informed of their task completion time, number of errors committed, and score. Accumulated points were used in competition for prizes.

Since the current work is focused on the effect of an automated cue and downstream error cost on postcompletion error, the experiment program allowed participants to complete a trial at testing without executing the postcompletion step (Tactical task), although a warning beep was emitted. The cued version of the same task featured a simple red visual cue appearing adjacent to the "Tracking" button at the postcompletion step on every trial (see Figure 3). In the mode error condition, the Tactical console stayed at the postcompletion step if it was forgotten on a previous trial and prevented the operator from proceeding until it was first completed. Finally, in the combined mode error and cued condition, the system combined both the downstream error cost of the mode error condition with the visual cue. Analysis of the results focused on the data collected from the testing day.

<< Figure 3 >>

2.3. Results

The mean frequency (out of all trials) of postcompletion errors across groups was of primary interest in our analysis. Participants who made an omission at the postcompletion step at more than 50% frequency were removed. This was due to the assumption that postcompletion errors occur in this case when the operator has the

correct plan. If errors at this step were shown to occur more than 50% of the trials, it is likely that the participant had not correctly remembered the task completely from training. Fifty percent was chosen as the point of delineation as participants tended to pool into two groups: those with postcompletion errors ranging from 38% frequency and below and those with postcompletion frequency at 70% frequency and above. Thirteen participants were found in the second category and removed: five from the control group, three from the mode error group, two from the cued group and three from the combined group. This left thirteen participants for the mode error and cued groups, twelve for the combined, and eleven for the control, combining for a total of forty-nine participants included in the final sample.

2.3.1. Postcompletion Frequency

Postcompletion frequency, or the number of postcompletion errors committed out of the total number of opportune steps (Frequency = Number of Errors at a Step X_i / Total Number of Opportunities for Error at Step X_i) was analyzed by condition (Figure 4). Contrary to the hypothesis, no reliable effect of group on postcompletion error frequency was found, $F(3, 45) = 1.22$, $p = 0.31$. The relatively low mean frequencies across groups may be indicative of floor effects due to the the removal of the participants with greater than 50% postcompletion error frequency. The obtained figures for postcompletion errors were low (e.g., 0.05 versus 0.11 for the Control condition) in comparison to the findings of Serig (2001). A power calculation for the ANOVA showed the effect size to be 0.25, providing a somewhat low power of 0.34 at the .05 level. Power to detect a medium sized effect (i.e., effect size of 0.4), however, was a somewhat more respectable 0.61, but this was hardly conclusive.

<< Figure 4 >>

2.3.2. *Postcompletion Step Times*

The average initial postcompletion step times reflect the difficulty faced by most of the participants on the first trial: 5736 ms (Control), 6911 ms (Cued), 8040 ms (Mode), and 5174 ms (Combined). This is the time taken from the postcompletion, or the second to last step in the task (which generates the false completion signal), and the actual last step in the task. Times declined overall for the rest of the trials, falling within the range found by Serig (~4500 ms, Day 2a; 2001). This may be explained by debriefing reports, in which participants stated that it was not very difficult to recall how to perform the tasks after the first trial.

A repeated-measures ANOVA revealed a reliable main effect of trial, $F(12, 504) = 7.323$, $p < .01$ and linear trend, $F(1, 31) = 28.15$, $p < .01$. There was also a reliable main effect of condition, $F(3, 42) = 8.23$, $p < .01$, although Tukey's HSD test showed none of the intervention conditions to be significantly different from the control, $p > .05$. Finally, there was no significant trial by condition interaction, $F(36, 504) = 1.23$, $p = .169$. Differences in task completion times were also unreliable between conditions, $p > .05$.

2.4. Discussion of Experiment 1

Despite the lack of reliable differences in reaction times or error commission at the postcompletion step, the results put forth some valuable implications. First, the fact

that the visual cue did not significantly reduce the number of postcompletion errors committed by the participants demonstrates that simply following a design heuristic and placing a contiguous and proximal reminder can be ineffective. While the sudden onset of a large red dot against a black and white console and next to the button to be pressed seemed intuitive and highly visible, participants made errors at the postcompletion step regardless. Second, returning to the issue of salience, it remains to be determined whether this is a problem at the level of vision, attention, or memory. Seemingly, they overlooked the cue or forgot its association with the postcompletion action of pressing the "Tracking" button. Critical observation of the system's state communicated via cues on the interface (text message and graphics) should have prompted them to consider the novel appearance of the visual cue next to the "Tracking" button. However, a high working memory load and a self-generated speed-accuracy tradeoff, encouraged by the time and performance pressures, may have allowed omissions to take place regardless.

Neither did the downstream error cost present a significant change in behavior at the postcompletion step for participants. It was hypothesized that this manipulation would act as negative feedback via the time cost added in response to an error, bringing about a change in behavior on subsequent trials. However, as the results show, it did not provide any significant change in participants' behavior. On some occasions the mode error condition incurred significant additional time cost at the initial step of the post-error trial. Nonetheless, this feedback did not seem to change participants' behavior or decrease their rate of error, contrary to our expectations. This perhaps follows findings by Serig (2001) that demonstrated participants' error commission to be relatively independent of negative or positive feedback about task performance.

The performance by the ten participants who were unable to recall the postcompletion step at above 50% frequency was also unexpected. The fact that ten of the initial forty-nine participants were unable to recall this task step is somewhat remarkable considering that they had all completed the extensive training session successfully. Nevertheless, despite a slightly positive relationship between the total number postcompletion errors and other errors, several of these participants with high numbers of postcompletion errors made relatively few other errors. This supports the notion that the postcompletion step is particularly difficult to remember due to the task characteristics provided by Reason (2002). In subjective reports, participants (including those who committed the error at over 50% frequency) reported that the task as a whole was generally not difficult to remember. However, some of those who did commit postcompletion errors at over 50% frequency noted that the task seemed to change or that they did not understand why the program kept beeping at the last step. Again this suggests that they had particular problems recalling the postcompletion step.

Among those participants initially removed from the data included several who failed to recall the postcompletion step altogether. Since the system no longer offered error messages telling the participant what the correct action was at each step, it is likely that as trials progressed with no strong negative or informative feedback from the system, those participants continued through the trials unaware of their mistake. The exception to this was the downstream cost condition, in which participants were unable to proceed on a subsequent trial without first completing the postcompletion step missed on the previous trial. Driven by the performance and time pressures, it is likely that they failed (or chose not to) to stop and consider their incorrect actions or attempt to recall the meaning of the cue. As these participants proceeded through the thirteen trials, that incorrect rule to ignore the cue or skip the postcompletion step may gain strength with repetition, making it increasingly difficult for the cue or the mode error to generate any further influence on their behavior. Another possible explanation for the unreliable differences between conditions is that the interventions lacked salience and/or participants forgot the association between the cue and postcompletion action. However, all of the participants read through a paper manual detailing the association of the cue to the postcompletion step and were not allowed to finish the training session until they could complete three further trials without the manual.

3. Experiment 2

The purpose of Experiment 2 was twofold: first, to improve the experimental procedure used in Experiment 1 and second, expand the breadth of inquiry to different types of cues. Specifically, cues that varied in appearance and function were investigated, according to evidence from previous work that suggested they would succeed. Previous issues with training, train-test delay length, salience of our interventions and strength of association, and the number of trials at testing were all considered in devising a follow-up study. A second task and interface was introduced under the fictional scenario of a Starfleet Chief Medical Officer training program to examine differences in the effects of the cues.

3.1. Intervention Implementation

Despite supporting theories and evidence suggesting that a simple visual cue would be effective as a reminder, the red onset used in Experiment 1 was not found to be effective. *This may be explained by Hollnagel's (1993, p.299)* assertion that the strength of a cue is relative to its specificity. Hence, it is the cue's strength relative to the other elements of the task that is important when assessing a cue's potential as a reminder. This claim is based on the observation that when a task is considered trivial, attention is more easily diverted. Performance becomes controlled by more error-prone generic functions such as "look for cue which indicates a turn," rather than exact intentions such as "look for cue-X, then turn to the right." For this reason, it was important to design and train participants to visually specific cues demanding explicit actions rather than generic ones affording a wider range of meaning. Such generic cues can potentially lead to description errors or errors caused by multiple cues with ambiguous specification of the required action or state if used within complex systems (Norman, 1988). Such lack of specificity may have been the problem with the cue used in the previous experiment.

In a study by Monk (1986), auditory cues were used to drastically reduce the occurrence of mode errors. Keying-contingent sounds were used on a keyboard-based computer game to enhance feedback and draw the user's attention to a change in the system's mode. This worked well because the nature of mode errors is such that they generally occur when the user is unaware of the system's current mode and its consequences. Monk (1986) observed that display changes, however, are effective when the user is required to look at the relevant parts of the display at the appropriate moment *in the dialogue. Pointing devices such as the mouse force users to focus on the screen,* making small visual changes or cues, which may go unnoticed with other types of interaction, more likely to be effective.

3.2. Cue Attributes

Evidence suggests that the visual attributes most effective for attracting attention (warnings and indicators) on a computer interface in order are as follows (Sutcliffe, 1995):

1. Movement (blinking or change of position)
2. Shape and Size (character font, shape of symbols, text size, size of symbols)
3. Color
4. Brightness
5. Shading and Texture (different texture or pattern)
6. Surroundings (borders, background color)

Sutcliffe (1995) suggests that care be taken to ensure that the user population interprets the warning icon or cue as the designer expects. Furthermore, such attributes should only be *applied sparingly, as the presence of many conflicting stimuli can essentially dull their* individual effectiveness.

For color, red, green, and yellow are recommended as status indicators, each corresponding to its meaning on a traffic light. To draw attention, white, yellow, and red are most effective, although yellow offers the best visibility. Based on these recommendations from the literature and the failure of the cue in Experiment 1 to reduce postcompletion errors against the control condition, alternating red and yellow blinking arrows (see Figure 5) were used in both the Tactical task and the new Medical task. The directional shape of the cue pointing towards the button made it visually specific, while the blinking and colors made it conspicuous to an extreme against the black and white interface. To ensure these assumptions were reasonable, the cue was presented to several people in a pilot study to ensure proper placement and that people associated the cue with the required step.

<< Figure 5 >>

3.3. A Mode Indicator

In addition to the new cue, which appears just-in-time with the postcompletion step, a mode indicator condition was introduced to examine the effect of a cue appearing prior to the postcompletion step. The previously used Tactical interface of the Bridge Officer Qualification program was redesigned for this manipulation, in which visibility of the state change was enhanced. As shown in Figure 6, the mode indicator consisted of a green light appearing on the "Tracking" button, crosshairs showing in the targeting window, and the message "Tracking Mode Enabled" appearing in yellow. It was thought that the combination of these three novel features would be sufficient to remind the user that the system was in a distinct Tracking mode. When combined with the given if-then rule at training (i.e., "If you see a mode indicator light the system is on"), the presence of the mode indicator implies the necessary corresponding action of turning off the Tracking system. Once the participant finishes the intermediary steps and hit the "Tracking" button a second time, the green light and message turned off to indicate that the Tracking mode has ended. This style of mode indicator is commonly found in real-world devices, such as on automobile dashboards and television remote controls.

<< Figure 6 >>

All three conditions (control, cued, and mode) with the cue and mode indicator appearing at the "Main Display" button instead of at "Tracking" were also implemented in the new Chief Medical Officer Qualification program (Figure 7). This task was similar to the Tactical task (Table 1) in that it had a postcompletion step, as identified by Hierarchical Task Analysis (Kirwan & Ainsworth, 1992) with a real-world, built into the task structure device (Chung, Zhang, Johnson, & Patel, 2003).

<< Figure 7 >>

<< Table 1 >>

3.4. Method

Training in the previous experiment was thorough and detailed, using manuals to promote mental models of the system (Norman, 1988). However, as noted, there were still problems in Experiment 1 with several participants who failed to recall the postcompletion step at greater than 70% frequency at testing. Hence, the association between the system change (pre-postcompletion) step and maintenance (postcompletion) steps was further emphasized in Experiment 2, with four main changes:

1. The training manuals were revised to promote a stronger mental model of the using more detailed pictures, diagrams, and instructions.
2. The delay between training and testing was reduced to two days.
3. A paper-based quiz was issued at the end of training to ensure that participants associated the cue or mode indicator and the postcompletion step.
4. At testing the system reminded participants of the postcompletion step on the first trial, if they forgot.

3.4.1. Participants

Ninety-one undergraduate (42 female and 49 male) students from Rice University aged 18 to 35 participated for course credit in a psychology course and additional cash prizes ranging from $10 to $40.

3.4.2. Materials

The materials for this experiment consisted of a paper instruction manual for each of the four tasks (Navigation, Transporter, Tactical, and Medical), paper quizzes for the first day, Apple iMac computers running the Bridge Officer Qualification and Chief Medical Officer Qualification applications written in Macintosh Common Lisp, and a web-based general questionnaire.

3.4.3. Design

Experiment 2 used a two-factor design with task and intervention as variables. Task consisted of two conditions: Bridge Officer and Chief Medical Officer. Intervention consisted of three conditions: control (no intervention), visual cue (alternating red and yellow blinking arrows), and mode indicator (mode indication for the system state change). Participants were randomly assigned to one of the six groups. The primary dependent measure was the number of postcompletion errors made during the Tactical and Medical tasks. Other dependent measures of interest included response times at the postcompletion step and the overall number of errors per task.

3.4.4. Procedure

As in Experiment 1, participants were run in two sessions. The first session served as a training session using written documentation for each of the tasks. Order of training was randomized for every participant. Once participants successfully completed the training trial with the manual and logged three subsequent error-free trials, they were *asked to move on to the next task. Errors resulted in warning beeps and messages and* participants were returned to the main control to restart the task. This was to prevent participants from completing training without having gone through each of the tasks at

least four times with all steps done correctly and completely. When training was complete, they were reminded that they would be tested for prizes in two days and given a short quiz to ensure that they had an accurate mental model.

The second session consisted of the test trials for both the Tactical and Medical tasks. In random order, participants completed seventeen trials of their assigned postcompletion task (Tactical or Medical) and 11 trials for each of the two dummy tasks (Navigation and Transporter) for a total of 39 trials for the test day. The number of trials for the postcompletion task was increased from 13 in the first experiment to provide greater power. At testing, the experiment program emitted warning beeps on error commission to warn individuals but did not immediately return them to the main control as in training. Participants were encouraged to work both accurately and quickly by means of a scoring system, prizes, and an onscreen timer as in the first experiment. The same auditory working memory task from the previous experiment was also used in this experiment for all task conditions at testing.

3.5. Results

Only data from 82 of the original 91 participants were kept in the final analysis. The primary reason for this was participant failure to show up at their assigned testing date, but there were also a few cases of incorrectly saved data files. Our primary measure of interest was the frequency of errors at the postcompletion step in both tasks. This is the step immediately following completion of the main task goal. In contrast to Experiment 1, there were no participants with greater than 50% postcompletion error frequency, suggesting that the correct knowledge was imparted and carried over to testing day.

Outliers in the response time data greater or less than three standard deviations from each participant's mean were removed and replaced with their mean.

For the Tactical task, mean postcompletion error frequencies were 6.81%, 0% (exact), and 6.21% for the control, cued, and mode indicator conditions, respectively (Figure 8). Immediately apparent is that the new cue *completely* eliminated postcompletion errors. That is, participants made *zero* postcompletion errors in this condition. Participants made significantly less errors in the cued condition versus the control, $t(76) = 3.14$, p = .002, and versus the mode indicator group, $t(76) = 2.81$, $p =$ .006. In comparison, the mode indicator failed to produce reliable differences with the control group, $t(76) = .263$, $p = .793$.

<< Figure 8 >>

In the simpler Medical task, mean errors at the postcompletion step were very low: 0.82%, 0%, and 1.99% for the control, cue and mode indicator conditions, respectively. Again, none of the twelve participants in the Medical cued condition made a single postcompletion error in all seventeen of their trials. The same planned comparisons done on the Tactical task revealed no reliable differences across intervention.

Whether due to the number or nature of the steps, the mean postcompletion step completion time was drastically shorter in the Medical task compared to the Tactical, 4168 ms (Tactical) versus 1053 ms (Medical). An ANOVA showed this effect of task to be reliable, $F(1, 76) = 305.41$, $p < .001$. The overall effect of intervention on postcompletion step time was also reliable, $F(2, 76) = 4.32$, $p = .017$. However, the intervention by task interaction was not, $F(2, 76) = 2.86$, $p = .63$.

The average number of total errors was also found to be higher for the Tactical task than the Medical: 0.67 in the Tactical versus 0.28 in the simpler Medical task, $F(1, 76) = 14.60$, $p < .001$. Differences across intervention were not reliable, $F(2, 76) = 2.24$, $p = .113$, although it should be noted that the total number of errors was slightly higher for both the cue and mode indicator conditions in both tasks. Participants showed no reliable differences in working memory performance regardless of task $F(1, 76) = 3.47$, $p = .07$ or intervention, $F(2, 76) = 1.09$, $p = .342$.

3.6. Discussion of Experiment 2

As reported, all sixteen participants in the cued condition of the Tactical task exhibited perfectly error-free performance at the postcompletion step. The intervention was strikingly successful at capturing visual attention and cueing memory, completely eliminating errors at that step for all participants on all trials. In contrast, the control and mode indicator groups showed mean postcompletion error frequencies between six and seven percent. This finding is consistent with the overabundance of recommendations in the human factors literature against introducing modes into a system (e.g., Norman, 1988). Since the initial graphical change occurs significantly prior to the postcompletion step, the mode indicator condition in Experiment 2 places demands on prospective memory. This is the remembering and execution of delayed plans with no additional prompts at the time of intended retrieval (Guynn, McDaniel, & Einstein, 1998). According to Marsh and Hicks (1998), prospective memory performance decreases with increasing load on the executive resources, such as working memory. Hence, mode indicators, which are sometimes used as memory aids and reminders, are in fact susceptible to the same stressors they are meant to alleviate. In contrast, the visual cue appeared just-in-time, which according to the Altmann and Trafton (2002) model is a necessary condition for an effective reminder.

The Medical task was not effective as a parallel of the Tactical task, as it failed to generate sufficient error rates to truly prove useful for comparing the effects of the interventions. Interestingly, however, the cue also completely eliminated errors in the Medical task (versus .82% and 1.99% for the control and mode indicator conditions, respectively), as in the Tactical task. The cue's strong effect seems robust across tasks, even with the lower overall error rates.

There are several possible explanations for the former finding. First, the medical task was substantially shorter in length, taking participants nearly one quarter of the time taken to finish the Tactical task. Second, due to intrinsic differences in the nature of the task, the assumed postcompletion step may not have been a postcompletion step at all. In fact, it was found that the *last* step of the task (return to Main Control) generated more errors than the supposed postcompletion step in the control condition. This leads to a third possible reason: the simpler and flatter goal structure of the shorter Medical task. The longer, more complex Tactical task required the application of additional If-Then operators to complete the task and contains more transitions between related "clusters" of buttons and steps on the interface. Hence, following the Byrne and Bovair (1997) account, the increased susceptibility for postcompletion errors in the Tactical task may be explained by its greater demand on working memory.

4. A Computational Model

Cognitive architectures may help overcome the existing weaknesses in error prediction methods in two ways. First, they can provide a highly developed representation of both the human perceptual system and the external environment via either a model of the system or the actual system itself (e.g., Byrne & Kirlik, 2005). Recent inquiries into human error occurring in interactive tasks with a computer have frequently focused on the goal structure (e.g., Gray, 2000) or memory (e.g., Byrne & Bovair, 1997) for explanations. However, errors rooted in perceptual mechanisms must also be addressed, since computer-based tasks are highly visual. Just as failing to see a stop sign or a red light while driving may result in a car accident, similarly failing to see or misinterpreting some component of an interface can lead to serious consequences as well. To account for perception at the level of the computer interface, a systematic approach integrating all aspects of human cognition is required. The ACT-R architecture is designed to fit this criterion (Anderson et al., 2004).

Although both Experiment 1 and Experiment 2 utilized two other manipulations apart from the cue (i.e., downstream error cost and mode indicator), the present model was focused on the cued and control conditions from Experiment 2. The main independent variable in this line of work has been the error intervention. Byrne and Bovair (1997) have demonstrated that the working memory load imposed by the digit span task in these experiments affects performance, leading to postcompletion errors. The model took into consideration their hypothesis and results, which are in keeping with general findings of task performance degradation under situations of high cognitive load (e.g., Ruffel-Smith, 1979). The model did not, however, account for skill learning, which occured at the beginning of the experiments.

Traditionally, *symbolic* systems have modeled consistent errors and errors of commission by assuming certain rules are missing or fail to apply (Van Lehn, 1989). Symbolic systems have more difficulty, however, with occasional slips or intrusions (Norman, 1981). On the other hand, *connectionist* models, with their holistic computation style, are intended to reproduce human-like errors and graceful degradation of performance under noise or component failures. However, scaling up to computer-based procedural tasks like that used here has generally not been attempted with connectionist systems. ACT-R, being a *hybrid* system, is better suited than traditional symbolic systems in this case, because activation of chunks is spread through an association network (e.g., Altmann & Trafton, 2002).

The architecture consists of a set of perceptual-motor modules (e.g., motor and visual), declarative memory, procedural memory, buffers, and a pattern matcher that work together to model human-like cognition. Declarative memory in ACT-R is represented as *chunks* of knowledge, whereas procedural memory consists of If-Then rules termed *productions*. The pattern matcher detects chunks placed into the buffers by the modules and production rules are selected to fire serially. The subsymbolic half of ACT-R helps guide the selection of rules and the internal operations of the modules. Learning and errors, for example, depend heavily on these subsymbolic processes. By taking advantage of ACT-R's subsymbolic construct of activation, potential errors at each step in the task structure can be produced. The increased working memory load, in its

model representation, "steals" activation required to make a retrieval of the postcompletion subgoal.

Lebiére, Anderson, and Reder (1994) have already demonstrated ACT-R's capacity to model graded human error, traditionally considered a domain restricted to connectionist models. Their study required participants to dual-task, as in the experiments reported here. Participants performed a high-level cognitive task of solving simple linear algebra problems while concurrently memorizing a digit span. The ACT-R model reproduced errors of omission by utilizing a cutoff on the latency of memory retrievals – retrievals failed if a chunk did not have sufficient activation. Because chunk activation is noisy, the model was able to generate a pattern of error quantitatively similar to participant data. With the current model, a similar method was utilized to generate errors of omission at the postcompletion step.

The model was built only to perform the Tactical task. Although it has some abstractions, particularly in the non-postcompletion steps, the focus was the postcompletion step itself where two key productions generated errors. The first of these initiated a retrieval of the postcompletion step information, which included the visual coordinates for the button to be clicked. Using partial matching in ACT-R, the level of "similarity" between the two knowledge chunks for the postcompletion step and the last step and the one slot calling for their retrieval were adjusted such that sometimes the incorrect chunk was retrieved. The additional working memory load was simulated using dummy chunks representing state information placed in the goal buffer. These chunks, which can be considered as the digits in the digit span task, "stole" activation available for the retrieval of the chunk that produces the postcompletion action. With activation noise enabled, random retrievals of the incorrect (last) step were generated, leading to postcompletion errors.

In the cue condition, a representation of the cue was automatically "stuffed" into ACT-R's visual buffer upon appearance (a native property of ACT-R's vision module), *in turn triggering a production that retrieves the correct knowledge chunk for the* postcompletion step. The production was an IF-THEN type rule, dictating retrieval of the postcompletion goal in response to the red cue. This followed instructions in the training manuals, which asked subjects to complete the postcompletion step when the "red indicator" lit up. In this case the automatic capture of visual attention generated by the cue's sudden onset led to the correct procedural knowledge or production firing, eliminating potential errors as found with participants.

The postcompletion frequency found in Experiment 1 for the control condition was 4.9%. However, this was only after participants who committed the error with over 50% frequency were removed in adherance to the definition of postcompletion errors as known-knowledge-based. Postcompletion frequency was at nearly 25% with their data included. In Experiment 2 the baseline postcompletion frequency was slightly lower, although this time participants with 25%+ frequency were absent. Thus, as a *compromise, 5-15% was the target postcompletion error frequency for this model. This* seemed reasonable considering that the cued and downstream error groups' participants exhibited postcompletion errors at around 10%.

Running the model for 200 trials generated a 14.1% postcompletion error frequency in the control condition. In the cued condition the model responded correctly every time (0% postcompletion error frequency), as we found in Experiment 2. Once

visual attention was "captured" by the novel appearance of the cue at the step, retrieval of the correct knowledge chunk for the postcompletion step was assured. This guaranteed retrieval resulted from the assumption that the cue induced accurate recall of procedural knowledge from training. The model thus demonstrates that if a just-in-time cue is *immediately meaningful and visually salient, correct performance should reliably result.*

The importance of meaningfulness in graphical user interface design is largely recognized in the applied world, as its presence (in some variation) on many usability guidelines and checklists indicates. It is an inherently variable property, however, since it is contingent upon preexisting user knowledge. Similarly, our model required knowledge of the cue to exist in order to interpret its appearance and trigger the appropriate production rule for the postcompletion step. The cue in Experiment 2 may have been particularly effective because its interpretation relied on strong, pre-existing knowledge that most people today have regarding arrow-shaped, blinking objects. This is in contrast to the cue in Experiment 1, which lacked any distinctive visual features and thus relied on less reliable knowledge from the one time training session.

5. General Discussion

These findings demonstrate the basic challenge of error prediction and remediation in dynamic computer-based tasks, which tend to be highly visual. Using "expert" knowledge, heuristics, or current error identification methods alone, it would be difficult to predict the disparity found between the two cues of Experiments 1 and 2 or even the mode indicator. Given that both cues appeared in proximity to the button and at the same time and both took advantage of color and novel appearance, the unaided evaluator would be hard-pressed to determine the extent to which one cue would be effective over the other using any simple heuristic. Returning to our initial hypotheses, there may be differences between the cues at the level of attention, perception, or memory. To direct attention, the two arrows used in Experiment 2 pointed distinctly at the button to be pressed. As Hollnagel (1993) recommends, cues should be specific in this manner to avoid ambiguity. In terms of perception, the arrows in Experiment 2 blinked and, thus, should have been more effective at capturing visual attention. Finally, the arrows relied on general, non-experiment specific knowledge in memory regarding the meaning of arrow-shaped objects. Any or all of these observations may explain why the cue in Experiment 2 was successful and the cue in Experiment 1 was not.

Furthermore, at the level of perception and attention, the flashing cue presents itself as repeated onsets, whereas the cue used in Experiment 1 appeared as a single onset. Sutcliffe (1995) notes that movement is the strongest visual attribute in determining whether or not a cue or warning is seen on the computer interface. With the Tactical task, participants must shift their visual attention from the targeting screen to the "Main Control" button in the case that they forget the postcompletion step. A message also appears at the bottom left hand side of the screen, in conjunction with auditory feedback, to inform the participant that the target has been destroyed. Should visual attention shift there and then immediately move to the Main Control button, a red onset appearing outside the new locus of attention could go unnoticed. As some research suggests (e.g., Mack & Rock, 1998), people may inhibit attention at a previous fixation when shifting attention to another area on the display. Feature-based models (Folk, Remington, & Johnston, 1992) of visual attention may also offer an explanation in that

people sometimes do not see "unexpected objects that have different properties from already-attended objects, even if the unexpected object appears in close proximity to the focus of attention," (p. 2, Most et al., 2000). Given a scenario in which the participant is in the process of making a postcompletion error, visual search is being guided top-down by an incorrect rule to find the black and white button labeled "Main Control," not a single red onset.

These results also fall in line with the predictions of Altmann and Trafton's MAGS (2002), assuming that the cue in the first experiment failed simply because it was not visually salient. In agreement with their model, a perhaps overly salient visual cue (Experiment 2) was sufficient to prime the postcompletion step, making it unnecessary to place the postcompletion action on the critical path. In contrast, the visually prominent mode indicator was unsuccessful at priming the subsequent postcompletion step, perhaps due to masking caused by the intermediate goal of firing the phaser. This presents further evidence of the significant role of working memory in human error.

The Medical task clearly illustrated the difficulty of designing a task that can elicit sufficient error rates from participants to study human error in the laboratory. Although it includes a task step after the main goal of the task, this supposed postcompletion step failed to generate significant error rates. This presents some implications for those doing formal evaluations of systems (e.g., heuristic evaluation), since it reveals the complexity of identifying potential error inducing steps or visual features of an interface. Simply conducting a task analysis or classifying errors using a taxonomy or list of usability guidelines, for instance, will not always be sufficient.

6. Conclusion

The visual design of an interface can greatly affect human error frequency in a computer-based routine procedural task. Both the mode indicator (which appears before the postcompletion step) in Experiment 2 and the downstream error cost in Experiment 1 had no reliable effect on postcompletion error frequencies, in contrast to the just-in-time cue introduced in Experiment 2, which completely eliminated the error. Furthermore, when implementing visual cues as reminders for users, the data demonstrate that movement and immediate meaningfulness (and/or specificity) are strong determinates of their effectiveness. This follows longstanding human factors guidelines (e.g., U.S. Department of Defense, 1999) and research (Wang, Cavanagh, & Green, 1994), which advise that indicators on an interface should be both visually salient and specific in their message. The cue in Experiment 1 appeared at the same location and point in time as the cue used in the second experiment, yet generated no reliable differences from the no cue control condition. The mode indicator, which relied on static contextual cues on the interface, also did not reduce the number of errors at the postcompletion step. Even negative feedback from the system (Experiment 1) in the form of a downstream error cost and encouragement from an "overseer" (Serig, 2001) failed to generate any significant reduction in the frequency of errors. Postcompletion errors cannot simply be willed away.

However, combatting potential errors may not be so simple as merely adding attention-grabbing cues to the interface as reminders. Differences in the design of the interface (e.g., background color) may also attenuate the visibility and, thus, effectiveness

of these cues. For example, the color of a cue may be affected by its relative contrast to a background color, as much of the visual attention literature indicates. Sutcliffe (1995) notes that the presence of more than one stimulus in conflict with the others can also reduce individual effectiveness. Additionally, the fact that our participants had explicit training on the meaning of the cue must be considered. Simply placing blinking arrows or other novel cues on the interface might affect new users differently from those who had been trained. As the model aptly demonstrates, the automatic capture of visual attention by the cue is insufficient to make a successful reminder, as knowledge is also necessary for correct interpretation.

This work is an initial step towards extending our knowledge of understanding how visual cues can be used as reminders in computer-based tasks. By examining how such interventions encourage correct behavior, we may improve our understanding of what cognitive factors lead to errors. While human performance data, physiological knowledge, and design guidelines exist to suggest the visual properties of effective cues and reminders, they are insufficient to account for the highly dynamic and perceptual variables endemic to computer-based tasks. Computer interfaces must be designed to both reduce the frequency of human error and remediate their effects, particularly in safety critical domains.

The traditional methods of task representation and error identification have been shown here to be inadequate for predicting low frequency errors in the highly visual domain of human-computer interaction (see also Byrne et al., 2004). As demonstrated by our initial modeling effort, understanding how the visual layout of an interface affects human performance requires consideration of perception, attention, and memory. A more extensive method that can account for these components of human cognition is necessary for the reliable prediction of errors and thorough interface evaluation in situations like that reported here. Such an approach would undoubtedly be unwieldy to implement manually and would also require significant amounts of data. Hence, some have suggested cognitive modeling as the eventual solution (see Byrne, 2003; Gray, 2004). Those studying eye movements in reading (e.g., Rayner, 1998) have similarly converted their massive amounts of data into a model that can be iteratively tested and validated. However, to successfully model error in computer-based tasks, it will first be necessary to continue this empirical investigation of how our visual system is guided by visual cues and features on the interface, building up an error "data bank" of sorts.

In conclusion, the prediction of human error in dynamic computer-based tasks is more difficult and complex than existing guidelines, theories, and evaluation techniques may lead one to believe. As demonstrated by the results from the two experiments, only extensive testing of the interface with human subjects allowed us to determine that one cue (Experiment 2) was effective and the other (Experiment 1) was not. Adapting these findings to a computational model provided an analytical tool applicable to a variety of interfaces and capable of producing quantitative predictions. In combination with other modeling work done to fit step completion times in this same task (Byrne et al., 2004), a more complete model that can yield both time and error predictions may eventually be generated. Such a model would be able to offer concrete explanations based on cognitive theory and specific design solutions for errors in interactive systems.

Acknowledgements:

We would like to acknowledge the support of the Office of Naval Research under grant number N00014-03-1-0094. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ONR, the U.S. Government, or any other organization.

References

Altmann, E. M. & Trafton, J. G. (1999). Memory for goals: An architectural perspective. *Proceedings of the Twenty-first Annual Conference of the Cognitive Science Society* (pp. 19-24). Hillsdale, NJ: Erlbaum.

Altmann, E. M., Trafton, J. G. (2002). Memory for goals: an activation-based model. *Cognitive Science, 26*, 39-83.

Anderson, J. R., & Lebiére, C. (1998). *The Atomic Components of Thought*. Mahway, NJ: Erlbaum.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Quin, Y. (2004). An integrated theory of the mind. *Psychological Review, 111*, 1036-1060.

Baber, C. (1996) Repertory grid and its application to product evaluation, In P.W. Jordan, B. Thomas, B.A. Weerdmeester & I. McClelland (Eds.) *Usability Evaluation in Industry*. London: Taylor and Francis 157-166

Baber, C. and Stanton, N. A. (1994) Task analysis for error identification. *Ergonomics 37 (11)*, 1923-1942

Byrne, M. D., & Bovair, S. (1997). A working memory model of a common procedural error. *Cognitive Science. 21(1)*, 31-61.

Byrne, M. D., & Kirlik, A. (2005). Using computational cognitive modeling to diagnose possible sources of aviation error. International Journal of Aviation Psychology, 15, 135-155.

Byrne, M. D., Maurier, D., Fick, C. S., & Chung, P. H. (2004). Routine procedural isomorphs and cognitive control structures. In C. D. Schunn, M. C. Lovett, C. Lebiere & P. Munro (Eds.), *Proceedings of the Sixth International Conference on Cognitive Modeling* (pp. 52-57).

Chung, P. H., Zhang, J., Johnson, T. R., & Patel, V. L. (2003). An extended hierarchical task analysis for error prediction in medical devices. *Proceedings of the American Medical Informatics Association (AMIA) Annual Symposium, American Medical Informatics Association, Washington D.C.*

Folk, C. L., Remington, R. W., & Johnston, J. C. (1992). Involuntary covert orienting is contingent on attentional control settings. *Journal of Experimental Psychology: Human Perception and Performance, 18*, 1030-1044.

Gray, W. D. (2000). The nature and processing of errors in interactive behavior. *Cognitive Science. 24(2)*, 205-248.

Gray, W. D. (2004). Errors in interactive behavior. In W. S. Bainbridge (Ed.), *Encyclopedia of Human-Computer Interaction* (pp. 230-235): Berkshire Publishing Group.

Green, M., & Senders, J. (2004). Human error in road accidents. *Visual Expert.* Retrieved March 27, 2005 from http://www.visualexpert.com/Resources/roadaccidents.html#Green91.

Guynn, M. J., McDaniel, M. A., & Einstein, G. O. (1998). Prospective memory: When reminders fail. *Memory & Cognition*, 26(2), 287-298.

Hollnagel, E. (1993). *Human reliability analysis, context and control*. Academic Press, London.

Hollnagel, E. (1998). *Cognitive reliability and error analysis method (CREAM)*. Oxford: Elsevier Science Ltd.

John, B. E., & Kieras, D. E. (1996). The GOMS family of user interface analysis techniques: Comparison and contrast. *ACM Transactions on Computer-Human Interaction, 3, 4*, 320-351.

Jonides, J. (1981). Voluntary versus automatic control over the mind's eye's movement. In J. B. Long & A. D. Baddeley (Eds.), *Attention and performance IX* (pp. 187-203). Hillsdale, NJ: Erlbaum.

Kieras, D. E., Wood, S. D., & Meyer, D. E. (1997). Predictive engineering models based on the EPIC architecture for multimodal high-performance human-computer interaction task. *ACM Transactions on Computer-Human Interaction, 4*, 230-275.

Kirwan, B. (1992). Human Error Identification in Human Reliability Assessment. *Applied Ergonomics*, 23(5), 299 – 381.

Kirwan, B. & Ainsworth, L. K. (Eds.) (1992). *A guide to task analysis*. London, UK: Taylor & Francis.

Lebiére, C., Anderson, J. R., & Reder, L. M. (1994). Error modeling in the ACT-R production system. *Proceedings of the Sixteenth Annual Meeting of the Cognitive Science Society* (pp. 555-559). Hillsdale, NJ: Erlbaum.

Mack, A., & Rock, I. (1998). *Inattentional blindness*. Boston, MA: MIT press.

Marsh, R. L., & Hicks, J. L. (1998). Event-based prospective memory and executive control of working memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 24(2)*, 336-349.

Monk, A. (1986). Mode errors: a user-centered analysis and some preventative measures using keying-contingent sound. *International Journal of Man-Machine Studies, 24*, 313-327.

Most, S. B., Simons, D. J., Scholl, B. J., Jimenez, R., Clifford, E., & Chabris, C. F. (2000). How not to be seen: The contribution of similarity and selective ignoring to sustained inattentional blindness. *Psychological Science, 12*, 9-17.

Norman, D. A. (1981). Categorization of action slips. *Psychological Review, 88*, 1-15.

Norman, D. A. (1988). *The design of everyday things*. New York: Basic Books.

Polson, P. G., & Lewis, C. H. (1990). Theory-based design for easily learned interfaces. *Human-Computer Interaction, 5*, 191-220.

Rasmussen, J. (1982). Human errors: a taxonomy for describing human malfunction in industrial installations. *Journal of Occupational Accidents, 4*, 311-335.

Rasmussen, J. (1987). The definition of human error and a taxonomy for technical system design. In J. Rasmussen, K. Duncan, & J. Leplat (Eds.), *New Technology and Human Error* (pp. 53–62). Chichester: John Wiley.

Rayner, K. (1998). Eye Movements in reading and information processing: 20 years of research. *Psychological Bulletin, 124*, 372-422.

Reason, J. (1990). *Human error*. Cambridge, UK: Cambridge University Press.

Reason, J. (2002). Combating omission errors through task analysis and good reminders. *Quality and Safety in Healthcare, 11*, 40-44.

Remington, R. W., Johnston, J. C., & Yantis, S. (1992). Involuntary attentional capture by abrupt onsets. *Perception & Psychophysics, 51*, 279-290.

Rieman, J., Young, R. M., & Howes, A. (1996). A dual-space model of iteratively deepening exploratory learning. *International Journal of Human-Computer Studies, 44*, 743-775.

Ruffel-Smith, H. P. (1979). A simulator study of the interaction of pilot workload with errors, vigilance, and decisions (Tech. Memo. 78482). NASA Ames Research Center: Moffett Field, CA.

Serig, E. M. (2001). *Evaluating Organizational Response to a Cognitive Problem: A Human Factors Approach.* Doctoral dissertation, Rice University, Houston, TX.

Stanton, N. A. (2004). Human error identification in human-computer interaction. In J. Jacko (Ed.), *Handbook of Human Factors and Ergonomics* (2^{nd} ed.). New York: John Wiley.

Sutcliffe, A. G. (1995). *Human-computer interface design.* London: Macmillan Press Ltd.

Swain, A. D., & Guttman, H. E. (1983). *Handbook of human reliability analysis with emphasis on nuclear power plant applications.* NUREG/CR-1278 (Washington D.C.).

Treisman, A. (1986) Features and objects in visual processing. *Scientific American, 254.* 114-124.

U.S. Department of Defense. (1999). Department of Defense Design Criteria Standard: Human Engineering (MIL-STD-1472F). Washington, DC: Government Printing Office.

Van Lehn, K. (1989). Problem solving and cognitive skill acquisition. In M. I. Posner (Ed.), *Foundations of cognitive science.* Cambridge, MA: MIT Press.

Wallace, D. F., & Huffman, D. (1990). Use of a visual cue to reduce errors in exiting a crash-bar type door. *Proceedings of the Human Factors Society* 34^{th} *Annual Meeting* (pp. 567-569).

Wang, Q., Cavanagh, P., and Green, M. (1994). Familiarity and pop-out in visual search. *Perception & Psychophysics. 56*, 495-500.

Wood, S.D. (2000). *Extending GOMS to human error and applying it to error-tolerant design.* Doctoral dissertation, University of Michigan.

Yantis, S., & Jonides, J. (1988). Uniqueness of abrupt visual onset in capturing attention. *Perception & Psychophysics, 43(4)*, 346-354.

Young, R. M., Barnard, P., Simon, T., & Whittington, J. (1989). How would your favorite user model cope with these scenarios? *ACM SIGCHI Bulletin, 20,* 51-55.

Zhang, J., & Norman, D. A. (1994). Representations in distributed cognitive tasks. *Cognitive Science, 18*, 87-122.

Figure 1: Photocopy hierarchical task structure. The ovals indicate major goals in the task while rectangles stand for subgoals. The dotted arrow indicates where the last associated goal (remove original) may be omitted to proceed to the next goal in an error situation.

Figure 2: Task hierarchy and screenshot of the Tactical task interface.

Figure 3: Visual cue appearing at the postcompletion step (Tracking button).

Figure 4: Postcompletion error frequencies by condition. Error bars indicate standard error of the mean.

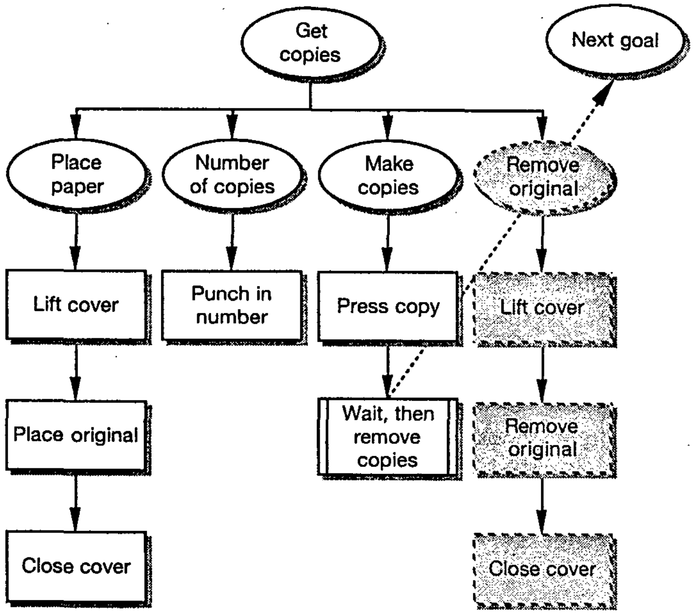
Figure 5: The new cue in the form of two blinking (red and yellow) arrows at the postcompletion step.

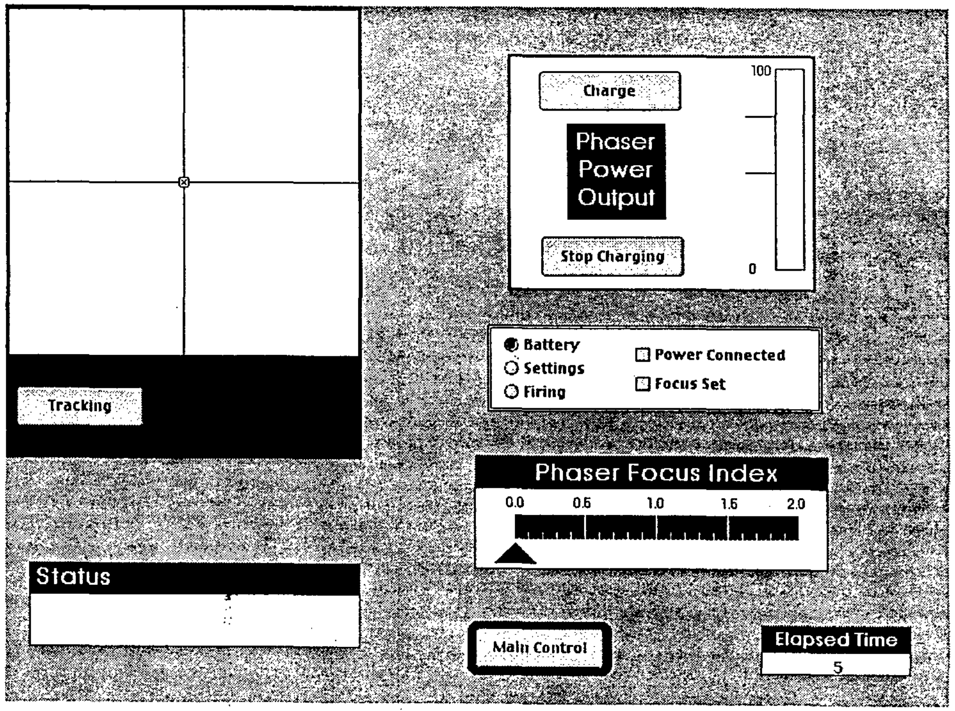
Figure 6: Mode Indicator in the Tactical task in on (panel a) and off (panel b) states.

Figure 7: Medical task interface in initial state.

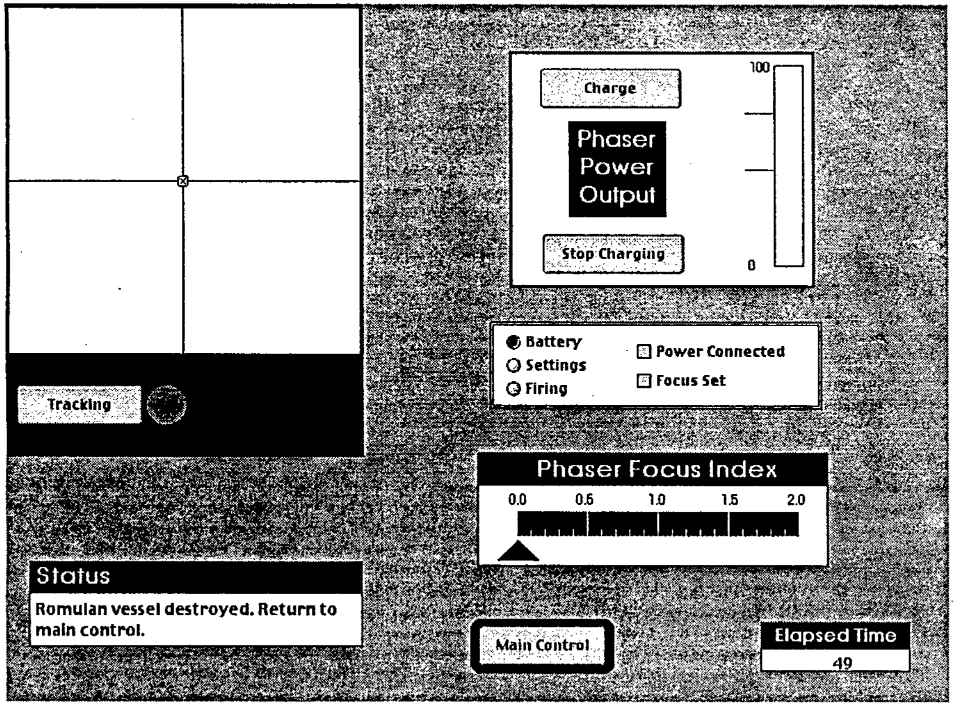
Table 1: Comparison of Tactical and Medical tasks.

Figure 8: PCE frequency by condition and task. Bars indicate standard error. Note. Cue is 0% in both tasks.





Goals	Steps
Charge Phaser	Power Connected
	Charge
	Stop Charging
	Power Connected
Set Focus	Settings
	<slider>
	Focus Set
Track Target	Firing
	Tracking
	<track>
Fire Phaser	Fire
	Tracking
next task	Main Control

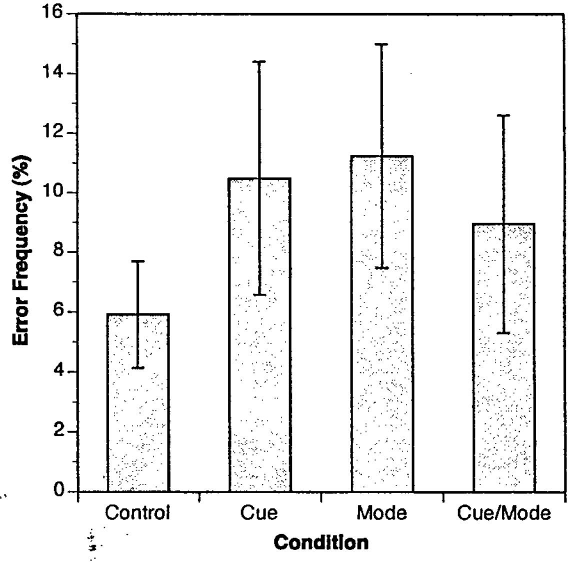


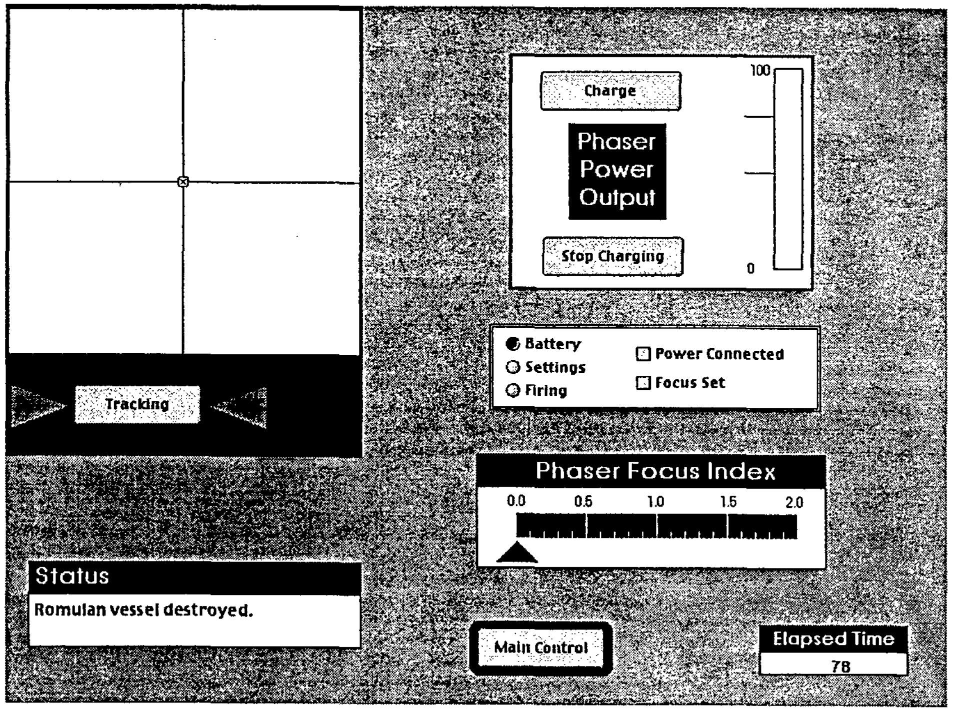
Status

Romulan vessel destroyed. Return to main control.

Elapsed Time

49

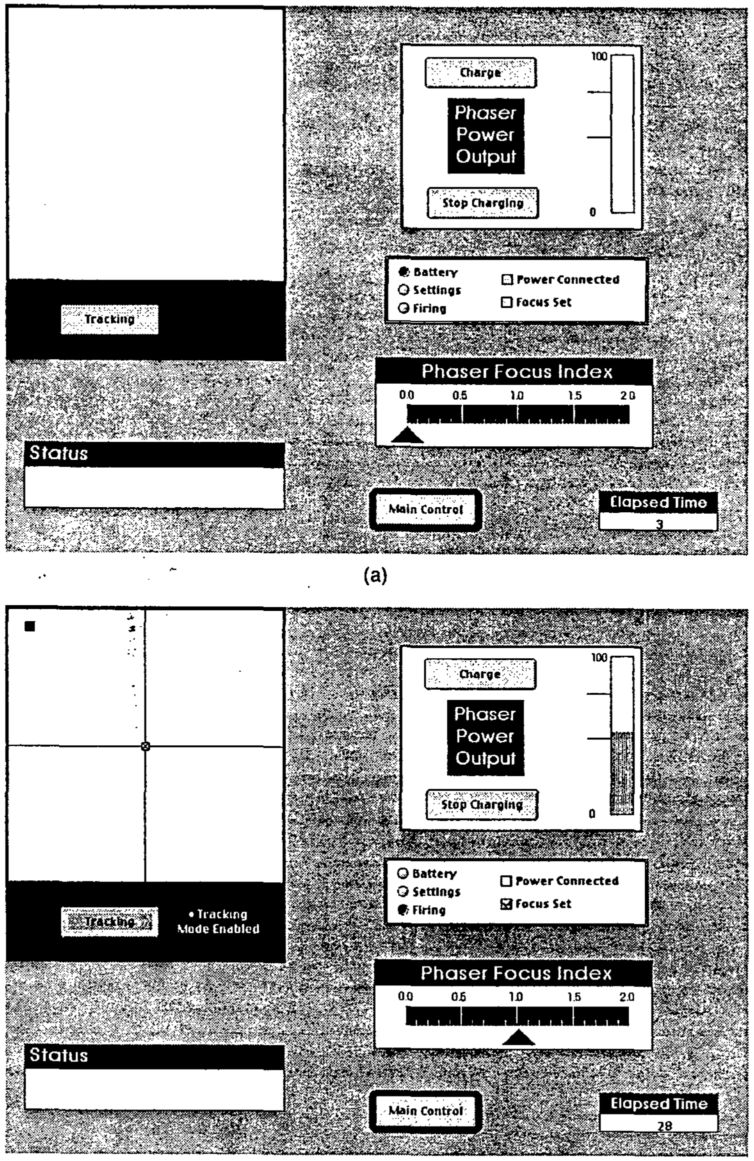




● Battery
○ Settings
○ Firing
□ Power Connected
□ Focus Set

Status
Romulan vessel destroyed.

Elapsed Time
78



● Battery
○ Settings
○ Firing
□ Power Connected
□ Focus Set

Status

Elapsed Time
3

(a)

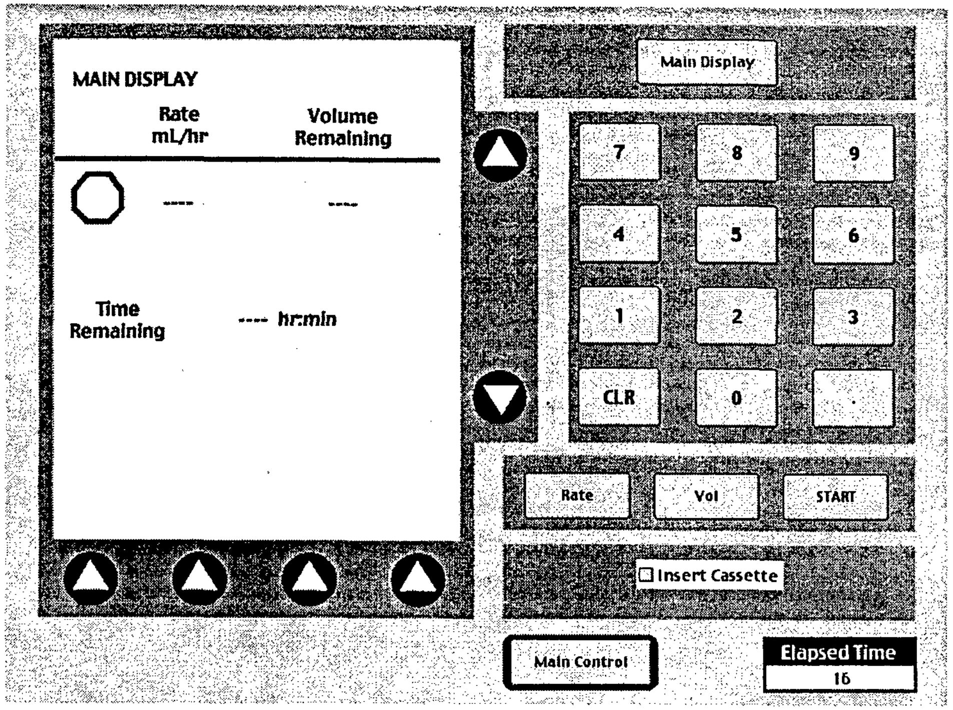
○ Battery
○ Settings
● Firing
□ Power Connected
☒ Focus Set

• Tracking Mode Enabled

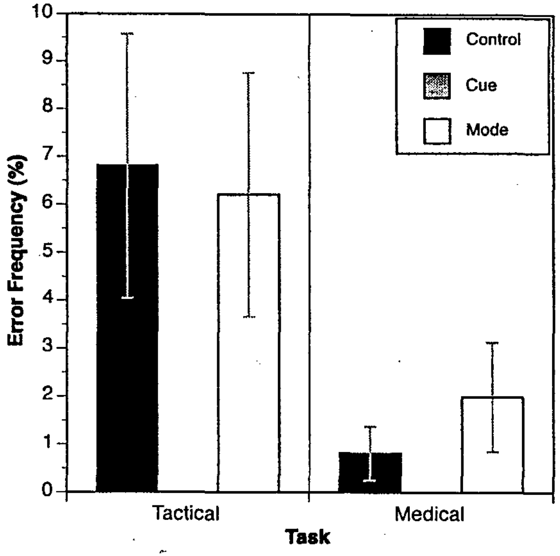
Status

Elapsed Time
28

(b)


MAIN DISPLAY
Rate
mL/hr
Volume
Remaining

Time
Remaining
---- hr:min
Main Display
7
8
9
4
5
6
1
2
3
CLR
0
.
Rate
Vol
START
□ Insert Cassette
Main Control
Elapsed Time
16


10
9
8
7
6
5
4
3
2
1
0
Error Frequency (%)
Tactical
Medical
Task
■ Control
▨ Cue
□ Mode

Tactical	Medical
Charge Phaser (5 substeps)	Insert Cassette (1 substep)
Set Focus (3 substeps)	Program Rate (2 substeps)
Track Target (3 substeps)	Program Vol (2 substeps)
Fire Phaser (4 substeps)	Start Flow (2 substeps)

Return to Main Control

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
23-02-2006

2. REPORT TYPE
Final technical report ✓

3. DATES COVERED *(From - To)*
22-OCT-2002 to 30-SEP-2005

4. TITLE AND SUBTITLE
Systematic Procedural Error

5a. CONTRACT NUMBER

5b. GRANT NUMBER
N00014-03-1-0094 ✓

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Michael D. Byrne

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

William Marsh Rice University
6100 Main St.
Houston, TX 77005

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
875 North Randolph St.
Arlington, VA 22217

10. SPONSOR/MONITOR'S ACRONYM(S)
ONR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for Public Release; Distribution is Unlimited ✓

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Even when executing routine procedures with relatively simple devices, people make nonrandom errors. Consequences range from the trivial to the fatal, with Navy personnel often operating at the more extreme end of this range. This problem has received surprisingly little attention from cognitive psychologists. The research summarized here examines such errors in some detail both empirically and through computational cognitive modeling. There were several key results. First, many such errors are sensitive not just to the structure of the task but also to the layout of controls and displays, contrary to the predictions of most current task analysis frameworks. Some such errors seem to be mitigable by simple layout changes. Second, a particularly pervasive error (termed postcompletion error) was found to be highly resistant to cue-based mitigation, and though an effective cue was found, the requirements for such cues are difficult to meet in field contexts. Finally, cognitive computational models constructed using the ACT-R cognitive architecture suggested that certain interface manipulations (removing state information, adding additional extraneous controls) which appeared major would actually have limited impact on human task performance, and these predictions were validated empirically.

15. SUBJECT TERMS
human error, cognitive psychology, visual attention, computational modeling

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	UU	23+appendices	Michael Byrne
UU	UU	UU			**19b. TELEPHONE NUMBER** *(include area code)* 713-348-3770